STATISTICALLY HIDING COMMITMENTS AND STATISTICAL ZERO-KNOWLEDGE ARGUMENTS FROM ANY ONE-WAY FUNCTION[*]

IFTACH HAITNER[†], MINH-HUYEN NGUYEN[‡], SHIEN JIN ONG[‡], OMER REINGOLD[§], AND SALIL VADHAN[‡]

Abstract. We give a construction of statistically hiding commitment schemes (ones where the hiding property holds against even computationally unbounded adversaries) under the minimal complexity assumption that one-way functions exist. Consequently, one-way functions suffice to give statistical zero-knowledge arguments for any NP statement (whereby even a computationally unbounded adversarial verifier learns nothing other than the fact the assertion being proven is true, and no polynomial-time adversarial prover can convince the verifier of a false statement). These results resolve an open question posed by Naor, Ostrovsky, Venkatesan, and Yung (CRYPTO '92, J. Cryptology '98).

Key words. cryptography, statistically hiding commitments, statistical zero-knowledge argument systems, one-way functions, interactive hashing

AMS subject classifications. 94A60, 68P25, 68Q99

1. Introduction.

As first discovered by Shannon [3] for the case of encryption, most interesting cryptographic tasks are impossible to achieve with absolute, information-theoretic security. Thus, modern cryptography aims to design protocols that are computationally intractable to break. Specifically, following Diffie and Hellman [4], this is typically done by showing that breaking the protocol is as hard as some intractable problem from complexity theory. Unfortunately, proving lower bounds of the sort needed seems beyond the reach of current techniques in complexity theory, and indeed would require at least proving P $\neq$ NP.

Given this state of affairs, research in the foundations of cryptography has aimed to design cryptographic protocols based on complexity assumptions that are as weak and general as possible. This project was enormously successful in the 1980's. In a beautiful sequence of works, it was shown that many cryptographic primitives, such as pseudorandom generators, pseudorandom functions, private-key encryption and authentication, digital signatures, (computationally hiding) commitment schemes, and (computational) zero-knowledge proofs could be constructed from any one-way function [5, 6, 7, 8, 9], and moreover this complexity assumption is minimal in the sense that each of these primitives (and indeed almost any cryptographic task) implies the existence of one-way functions [10, 11]. Moreover, it was shown that many of the remaining primitives, such as public-key encryption, collision-resistant hashing, and oblivious transfer, could not be reduced to the existence of one-way functions in a "black-box" manner [12, 13].

However, a few important primitives resisted classification into the above cat-

[*] Preliminary versions of this paper appeared as [1] and [2].

[†] Microsoft Research, New England Campus. E-mail: iftach@microsoft.com. Most of the research was done while at Weizmann Institute of Science, supported by grant no. 1300/05 from the Israel Science Foundation.

[‡] Harvard School of Engineering and Applied Sciences, Cambridge, Massachusetts, USA. E-mail: {mnguyen,shienjin,salil}@eecs.harvard.edu. Supported by NSF grant CNS-0430336, ONR grant N00014-04-1-0478, and US-Israel BSF grant 2006060.

[§] Weizmann Institute of Science, Rehovot, Israel. E-mail: omer.reingold@weizmann.ac.il. Research supported by grant no. 1300/05 from the Israel Science Foundation.

egories. That is, it was only known how to build these primitives from seemingly stronger assumptions than the existence of one-way functions, yet there was no black-box separation between these primitives and one-way functions. In this work, we are interested in two such examples — statistically hiding commitment schemes and statistical zero-knowledge arguments for NP.

1.1. Statistically Hiding Commitments. A commitment scheme defines a two-stage interactive protocol between a sender S and a receiver R; informally, after the ***commit stage***, S is bound to (at most) one value, which stays hidden from R, and in the ***reveal stage*** R learns this value. The two security properties hinted at in this informal description are known as ***binding*** (namely, that S is bound to at most one value after the commit stage) and ***hiding*** (namely, that R does not learn the value to which S commits before the reveal stage).

As with most cryptographic primitives, each of these security properties comes in two main flavors — ***computational security,*** whereby a polynomial-time adversary cannot violate the property except with negligible probability, and the stronger notion of ***statistical security***, whereby even a computationally unbounded adversary cannot violate the property except with negligible probability. (An even stronger notion is that of ***perfect security***, in which we do not even allow a negligible probability of breaking the scheme.) Naturally, statistical security, when achievable, is preferable to computational security. However, it can be shown that there do not exist commitment schemes that are simultaneously statistically hiding and statistically binding. Thus, at best we can hope for one of the two properties to be statistical and the other to be computational.

The complexity of *statistically binding* commitment schemes has been understood for a long time; they can be constructed from any one-way function [8, 5] and conversely, one-way functions are necessary for commitment schemes, even with both security properties computational [10]. In this work, however, we are interested in *statistically hiding* commitments, which have some advantages over statistically binding commitments. Specifically, when commitment schemes are used in constructing larger protocols, one typically needs the binding property to ensure the integrity of commitments that are opened *during* the protocol execution itself, and the hiding property to ensure that the unopened commitments remain secret even *after* the protocol execution. Thus, for the binding property, we need only be concerned with the adversary's current resources, and thus it may be safe for this property to be computational. For the hiding property, however, we need to consider resources that the adversary may gain far into the future, and thus statistical security is preferable.

Some of the most important examples of cryptographic protocols based on commitments are the zero-knowledge protocols for proving membership in an arbitrary NP language [9, 14]. In the protocol of [9], the hiding property of the commitment scheme translates to the zero-knowledge property of the protocol (i.e. the verifier learns nothing other than the fact that the assertion being proven is true), and the binding property of the commitment translates to the soundness property of the protocol, (i.e. the prover cannot convince the verifier of a false assertion). Thus, the existence of statistically hiding commitments implies that arbitrary NP statements can be proven with statistical zero knowledge and computational soundness; that is, every language in NP has a ***statistical zero-knowledge argument system*** [14, 15, 16].

Using statistically hiding commitments and the resulting statistical zero-knowledge arguments in known reductions [9, 17], one can actually transform *any* two-party protocol that is secure against passive (a.k.a. honest-but-curious) adversaries into one that is secure against malicious adversaries *while preserving statistical security for one of the two parties.*

Perfectly hiding commitment schemes and perfect zero-knowledge arguments for NP were first shown to exist based on specific number-theoretic assumptions [14, 18, 15, 19, 20] or, more generally, based on any collection of claw-free permutations [21, 22]. The assumption for statistically hiding commitment schemes and statistical zero-knowledge arguments was reduced further to collision-resistant hash functions [23, 24]. Even though it seems intuitive that the computational binding property of statistically hiding commitments should be closely related to collision resistance, the beautiful work of Naor, Ostrovsky, Venkatesan, and Yung [16] showed that actually any one-way permutation can be used to construct a perfectly hiding commitment schemes. Recently, Haitner et. al. [25] reduced the assumption further by constructing statistically hiding commitment based on regular one-way functions with known preimage size, and more generally on one-way functions where the preimage sizes can be efficiently approximated. The question of whether an arbitrary, unstructured one-way function implies statistically hiding commitments or statistical zero-knowledge arguments for NP, however, was left open.

1.2. Our results. In this paper, we resolve the complexity of statistically hiding commitments.

Theorem 1.1. *If one-way functions exist, then statistically hiding commitment schemes exist.*

By Impagliazzo and Luby [10], the existence of commitment schemes implies the existence of one-way functions and thus the above result is tight.

As discussed above, combining Theorem 1.1 and standard constructions of zero-knowledge protocols from commitments (cf., [9, 14, 15, 16, 26]), we obtain our second main result:

Theorem 1.2. *If one-way functions exist,*[1] *then every language in* NP *has a statistical zero-knowledge argument system.*

The assumption that one-way functions exist also seems to be essentially minimal here: Ostrovsky and Wigderson [27, 11] showed that a zero-knowledge argument system for a hard-on-average problem implies the existence of one-way functions, and it follows from [26] that a zero-knowledge argument system for a language outside of AM $\cap$ coAM (or even outside SZKP) implies the existence of "nonuniform" one-way functions, where both the efficiency and security refer to polynomial-sized circuits (and security holds for infinitely many input lengths).

To avoid a lengthy detour into zero knowledge, we omit the formal definitions and proofs needed for Theorem 1.2, and instead refer to [26], where our work plays a key role in proving *unconditional* results about zero-knowledge arguments.

1.3. Our techniques. We begin by using one-way functions to construct a variant of commitment schemes called *two-phase commitment schemes*, recently introduced by Nguyen and Vadhan [28].[2] We then use this two-phase commitment scheme together with *universal one-way hash functions* (whose existence is also implied by

[1] The standard definitions of zero-knowledge and soundness are nonuniform notions of security, and thus this theorem requires assuming the existence of one-way functions that are secure even against nonuniform adversaries.

[2] Using methods from [28], one can directly construct statistically zero-knowledge arguments for NP from two-phase commitments, and thereby prove Theorem 1.2. However, it is conceptually simpler to prove Theorem 1.1 and then deduce Theorem 1.2 using standard constructions.

the existence of one-way functions [7]) to construct the desired statistically hiding commitment scheme.

1.3.1. Two-phase commitments from any one-way function. Two-phase commitment schemes are commitment schemes with two phases, each consisting of a commit stage and a reveal stage. In the first phase, the sender commits to and reveals one value v_1, and subsequently, in the second phase, the sender commits to and reveals a second value v_2. We say that the two-phase commitment is ***hiding*** if both phases are hiding, and say that it is ***1-out-of-2-binding***, symbolically written as $\binom{2}{1}$***-binding***, if the following holds: with high probability, the sender will be forced to reveal the correct committed value in at least one of the phases (but which of the two phases can be determined dynamically by the malicious sender). More specifically, with high probability after the first-phase commit, there is a *single* value such that if the sender decommits to any other value, then the second commitment is guaranteed to be binding (in the standard sense).

Even though we draw upon [28] for the notion of two-phase commitments, there are many differences between the contexts of the two works and their constructions of two-phase commitments. In [28], the goal was to prove *unconditional* results about prover efficiency in zero-knowledge proofs (specifically, that one can transform zero-knowledge proofs with inefficient provers into ones with efficient provers). This was done by showing that every problem having a zero-knowledge proof has an "instance-dependent" two-phase commitment scheme, where the sender and receiver get an instance x of the problem as auxiliary input and we only require hiding to hold when x is a "yes instance" and binding when x is a "no instance." Here, we are giving *conditional* results (assuming the existence of one-way functions) and are obtaining standard (as opposed to instance-dependent) two-phase commitments. Moreover, the focus in [28] is on *statistically binding* two-phase commitments; thus here we need to develop new formulations to work with the computational binding property.

Our initial construction, which gives a two-phase commitment scheme satisfying a "weak hiding" property, is inspired by the construction of [28]. Indeed, the second phase in [28] was also introduced to deal with non-regular functions (corresponding to "non-flat distributions" in their setting), and our construction can be seen as applying the same idea to a variant of the protocol of [25]. However, in [28], this construction immediately gives a "strong hiding" property, whereas much of the technical work in the current paper comes from amplifying the "weak hiding" property we obtain into a strong one.

Like [28], another complication is that our initial construction does not provide a single two-phase commitment scheme, but rather polynomially many schemes, one of which is weakly hiding and all of which are $\binom{2}{1}$-binding. After applying our amplification procedure and the transformation discussed below to each of these schemes, we obtain polynomially many standard commitments, at least one of which is statistically hiding and all of which are computationally binding; these can then be combined into a single statistically hiding commitment scheme using standard techniques.

1.3.2. From 1-out-of-2-binding commitments to standard commitments. We would like to use a two-phase commitment schemes to construct a (standard) commitment scheme. A naive attempt would simply have the receiver randomly choose, after the first commit phase, whether to stick with the first-phase commitment or to use the second-phase as the actual commitment instead. The intuition is that since the commitment is $\binom{2}{1}$-binding, the sender cannot cheat in both phases together and thus the receiver would catch a cheating sender with probability half (which we can then boost using standard techniques). The problem is, however, that the sender can decide in which phase he will cheat *after* knowing the receiver's choice. Hence, the sender can cheat successfully in both cases without violating the $\binom{2}{1}$-binding of the underlying protocol.

Our additional idea is to use a *universal one-way hash function* in order to force the sender to decide in which phase it is about to cheat *before* knowing the receiver's choice. Universal one-way hash functions are a relaxation of collision-resistant hash functions that were defined by Naor and Yung [23] and shown to be constructible from any one-way function by Rompel [7]. (See also [29].) We show that the above problem can be solved by having the sender provide a universal one-way hash of the secret he has committed to in the first phase. This turns out to (computationally) determine whether the first or second phase will be binding while leaving enough entropy in the first-phase secret to still achieve hiding.

1.4. Subsequent Work. As the above suggests, our construction and its analysis are rather involved. Fortunately, a much simpler and more direct construction has recently been found [30]. Some of the techniques we develop here (such as those for working with collision probability as a measure of hiding in Section 6) may nevertheless still be useful for other purposes.

1.5. Outline. We present the basic notations and definitions in Section 2. As a warm up, we present constructions of statistically hiding commitments based on one-way permutations in Section 3 and from *regular* one-way functions with *known preimage size* in Section 4. In Section 5, we show how to construct two-phase commitments from regular one-way functions with *unknown preimage size* and in Section 6, we extend it to *any* one-way function. Finally, in Section 7, we present our transformation from two-phase commitments to (standard) statistically hiding commitments.

2. Definitions.

2.1. Basic notation. If X is a random variable taking values in a finite set $\mathcal{U}$, then we write $x \overset{\mathrm{R}}{\leftarrow} X$ to indicate that x is selected according to X. If S is a subset of $\mathcal{U}$, then $x \overset{\mathrm{R}}{\leftarrow} S$ means that x is selected according to the uniform distribution on S. When the universe $\mathcal{U}$ is clear from context, we write $\mu(S) = |S|/|\mathcal{U}|$ to denote the density of S. We adopt the convention that when the same random variable occurs several times in an expression, all occurrences refer to a single sample. For example, $\Pr[f(X) = X]$ is defined to be the probability that when $x \overset{\mathrm{R}}{\leftarrow} X$, we have $f(x) = x$. We write U_n to denote the random variable distributed uniformly over $\{0,1\}^n$. The ***support*** of a random variable X is $\mathrm{Supp}(X) = \{x : \Pr[X = x] > 0\}$. A random variable is ***flat*** if it is uniform over its support. If X and Y are random variables, then $X \otimes Y$ denotes the random variable obtained by taking independent random samples $x \overset{\mathrm{R}}{\leftarrow} X$ and $y \overset{\mathrm{R}}{\leftarrow} Y$ and outputting the pair (x, y). We write $\otimes^k X$ to denote the random variable consisting of k independent copies of X. For an event E, $X|_E$ denotes the random variable X conditioned on E. The ***statistical difference*** (also known as the ***variation distance***) between random variables X and Y taking values in $\mathcal{U}$ is defined to be $\Delta(X,Y) = \max_{S \subset \mathcal{U}} |\Pr[X \in S] - \Pr[Y \in S]|$. We say that X and Y are ε***-close*** if $\Delta(X,Y) \le \varepsilon$.

A function $\varepsilon : \mathbb{N} \to [0,1]$ is called ***negligible*** if $\varepsilon(n) = n^{-\omega(1)}$. We let $\mathrm{neg}(n)$ denote an arbitrary negligible function (i.e., when we say that $f(n) < \mathrm{neg}(n)$ we mean that *there exists* a negligible function $\varepsilon(n)$ such that for every n, $f(n) < \varepsilon(n)$). Likewise, $\mathrm{poly}(n)$ denotes an arbitrary function $f(n) = n^{O(1)}$.

For a probabilistic algorithm A, we write $A(x; r)$ to denote the output of A on input x and coin tosses r. In this case, $A(x)$ is a random variable representing the output of A for uniformly selected coin tosses. ***PPT*** refers to probabilistic algorithms (i.e., Turing machines) that run in *strict* polynomial time. A ***nonuniform*** PPT algorithm is a pair $(A, \bar{z})$, where $\bar{z} = z_1, z_2, \ldots$ is an infinite sequence of strings in which $|z_n| = \text{poly}(n)$, and A is a PPT algorithm that receives pairs of inputs of the form $(x, z_{|x|})$. (The string z_n is called the ***advice string*** for A for inputs of length n.) Nonuniform PPT algorithms are equivalent to (nonuniform) families of polynomial-sized Boolean circuits.

Two probability ensembles $\{X_n\}_{n\in\mathbb{N}}$ and $\{Y_n\}_{n\in\mathbb{N}}$ are ***computationally indistinguishable*** if for every PPT D, there exists a negligible function ε such that for all $n \in \mathbb{N}$,

$$|\Pr[D(1^n, X_n) = 1] - \Pr[D(1^n, Y_n) = 1]| \leq \varepsilon(|x|).$$

Similarly, we say that $\{X_n\}$ and $\{Y_n\}$ are ***statistically indistinguishable*** if the above is required for all functions D (instead of only PPT D's). Equivalently, $\{X_n\}$ and $\{Y_n\}$ are statistically indistinguishable if X_n and Y_n are $\varepsilon(n)$-close for some negligible function ε and all $n \in \mathbb{N}$.

An ***interactive protocol*** (A, B) consists of two algorithms that compute the next-message functions of the (honest) parties in the protocol. Specifically,
$A(x, a, \alpha_1, \ldots, \alpha_k; r)$ denotes the next message α_{k+1} sent by party A when the common input is x, A's auxiliary input is a, A's coin tosses are r, and the messages exchanged so far are $\alpha_1, \ldots, \alpha_k$. There is a special message, `halt`, which immediately halts the interaction, at which time each party can compute one more message, which is their ***private output***. Sometimes we will refer to protocols with a ***joint output***, which is required to be a deterministic polynomial-time function of just the common input and transcript of messages exchanged (and not the parties' auxiliary inputs or private coin tosses). We say that party A (resp., B) is ***probabilistic polynomial time (PPT)*** if its next-message function can be computed in polynomial time (in $|x| + |a| + |\alpha_1| + \cdots + |\alpha_k|$).

For an interactive protocol (A, B), we write $(A(a), B(b))(x)$ to denote the random process obtained by having A and B interact on common input x, with (private) auxiliary inputs a and b to A and B, respectively (if any), and with independent random coin tosses for A and B. We call (A, B) ***polynomially bounded*** if there is a polynomial p such that for all x, a, b, the total length of all messages exchanged in $(A(a), B(b))(x)$ is at most $p(|x|)$ with probability 1. Moreover, if B^* is any interactive algorithm, then A will immediately halt in $(A(a), B^*(b))(x)$ if the total length of the messages ever exceeds $p(|x|)$; we have the analogous requirement for B interacting with any A^*. We call a protocol (A, B) a ***polynomial time*** if (A, B) is polynomially bounded and both A and B are PPT.

The number of ***rounds*** in an execution of the protocol is the *total* number of messages exchanged between A and B, not including the final `accept/reject` message. We call the protocol (A, B) ***public coin*** for A (resp., B) if all the messages sent by A (resp., B) are simply the output of its coin tosses (independent of the history), except for the final `halt` message and A's (resp., B's) private output, which is computed as a deterministic function of the transcript.

We associate several random variables with the interaction $(A(a), B(b))(x)$. The private output of A is denoted by $\text{output}_A(A(a), B(b))(x)$, and $\text{view}_A(A(a), B(b))(x)$ denotes A's ***view*** of the interaction, i.e., its values are transcripts $(\gamma_1, \gamma_2, \ldots, \gamma_t; r)$, where the γ_i's are all the messages exchanged and r is A's coin tosses. Similarly, $\mathrm{output}_B(A(a), B(b))(x)$ and $\mathrm{view}_B(A(a), B(b))$ denote B's private output and view, respectively. The joint output, if any, is denoted by $\mathrm{output}(A(a), B(b))(x)$.

2.2. One-way functions. The most basic primitive of modern cryptography is a one-way function, which is a function that is *easy to compute* but *hard to invert.*

Definition 2.1. *Let* $s\colon \mathbb{N} \to \mathbb{N}$ *be any function. A function* $f\colon \{0,1\}^* \to \{0,1\}^*$ *is a* $s(n)$**-secure one-way function**, *or equivalently* **has security** $s(n)$*, if* f *is computable in polynomial time and for every PPT* A,

$$\Pr_{y \leftarrow \{0,1\}^n}[A(1^n, f(y)) \in f^{-1}(f(y))] < 1/s(n),$$

for all sufficiently large n*. Function* f *is a* **one-way function** *if* f *is* $s(n)$*-secure for every polynomial* s*. If the above holds also for nonuniform PPT* A*, we say that* f *is* **nonuniformly** *secure.*

One-way function f is a ***regular one-way function*** if there exists a function $g\colon \mathbb{N} \to \mathbb{N}$ such that $\forall z \in \mathrm{Supp}(f(U_n))$, $|\{y \in \{0,1\}^n : f(y) = z\}| = g(n)$; $g(n)$ is called the ***preimage size*** of f. We say that f is ***known-regular*** if $g(n)$ can be computed in time $\mathrm{poly}(n)$.

Without loss of generality, we can consider only one-way functions (regular or non-regular) that are length-preserving, that is for all $y \in \{0,1\}^*$, $|f(y)| = |y|$. This is because general one-way functions can be converted into ones that are length-preserving (cf., [31, p. 39]).

2.3. Commitment schemes. Another basic primitive of modern cryptography is a ***(bit) commitment scheme***, which is a two-stage protocol between a sender and a receiver. In the first stage, called the ***commit stage***, the sender *commits* to a private bit b. In the second stage, called the ***reveal stage***, the sender reveals b and *proves* that it was the bit to which she committed in the first stage. We require two properties of commitment schemes. The ***hiding*** property says that the receiver learns nothing about b in the commit stage. The ***binding*** property says that after the commit stage, the sender is bound to a particular value of b; that is, she cannot successfully open the commitment to two different bits in the reveal stage.

Definition 2.2. *An* **commitment scheme** *is a polynomial-time interactive protocol* $\mathsf{Com} = (S, R)$ *with the following properties:*

1. *Scheme* Com *proceeds in two stages: a* **commit stage** *and a* **reveal stage***. In both stages, the* **sender** S *and the* **receiver** R *receive a security parameter* 1^n *as common input.*
2. *At the beginning of the commit stage, sender* S *receives a private input* $b \in \{0,1\}$*, which denotes the bit that* S *is supposed to commit to. The commitment stage results in a joint output, which we call the* **commitment** $c = \mathrm{output}((S(b), R)(1^n))$*, and a private output for* S*, which we call the* **decommitment** *string* $d = \mathrm{output}_S(S(b), R)(1^n)$*. Without loss of generality,* c *can be taken to be the full transcript of the interaction between* S *and* R*, and* d *to be the private coin tosses of* S*.*
3. *In the reveal stage, sender* S *sends the pair* (b, d)*, where* d *is the decommitment string for bit* b*. Receiver* R *accepts or rejects based on* b*,* d*, and* c*.*
4. R *will always accept (with probability 1) if both sender* S *and receiver* R *follow their prescribed strategy.*

A commitment scheme is **public coin** *if all messages sent by the receiver are independent random coins.*

Next, we define the hiding and binding properties of commitment schemes.

Definition 2.3. *Commitment scheme* $\mathsf{Com} = (S, R)$ *is* **statistically [resp., computationally] hiding** *if for every [resp., PPT]* R^**, the ensembles* $\{\text{view}_{R^*}(S(0), R^*)(1^n)\}_{n\in\mathbb{N}}$ *and* $\{\text{view}_{R^*}(S(1), R^*)(1^n)\}_{n\in\mathbb{N}}$ *are statistically [resp., computationally] indistinguishable, where* $\text{view}_{R^*}(S(b), R^*)$ *denotes the view of* R^* *in the commit stage interacting with* $S(b)$*.*

Definition 2.4. *Commitment scheme* $\mathsf{Com} = (S, R)$ *is* **statistically [resp., computationally] binding** *if for every [resp., PPT]* S^**, there exists a negligible function* ε *such that the malicious sender* S^* *succeeds in the following game with probability at most* $\varepsilon(n)$*:*

> *On security parameter* 1^n*,* S^* *interacts with* R *in the commit stage obtaining commitment* c*. Then* S^* *outputs pairs* $(0, d_0)$ *and* $(1, d_1)$*, and* succeeds *if in the reveal stage,* $R(0, d_0, c) = R(1, d_1, c) = \texttt{accept}$*.*

If the above holds for every nonuniform PPT S^**, we say that* Com *is* **computationally binding with nonuniform security***.*

Constructing commitment schemes based on any one-way function. Naor [8] constructed commitment schemes that are computationally hiding and statistically binding from any *pseudorandom generator*, which in turn can be based on any one-way function [5]. The main result of this paper, Theorem 1.1, shows that commitment schemes that are *statistically hiding* and computationally binding can be based on any one-way function.

3. Statistically Hiding Commitments From One-Way Permutations.

Consider a one-way permutation $f\colon \{0,1\}^n \to \{0,1\}^n$. Naor, Ostrovsky, Venkatesan, and Yung [16] obtained a statistically hiding commitment scheme based on f by using a protocol called *interactive hashing* as a subroutine. Our agenda for this section is as follows: we will first informally describe interactive hashing and state the two main properties that we want from it; then, in Section 3.1 we give an informal description of the Naor et al. [16] scheme, henceforth called the NOVY commitment scheme; and finally, in Section 3.2, we give a formal definition of interactive hashing and a protocol satisfying that definition.

Interactive hashing is a protocol between a sender S_{IH} and receiver R_{IH}. The sender begins with a private input z, and at the end both parties output z_0 and z_1 such that $z \in \{z_0, z_1\}$. Informally, an interactive hashing protocol has the following two properties:

1. *Hiding*: if the sender's private input z is uniformly random, then every receiver, even computationally-unbounded malicious ones, does not learn which of z_0 or z_1 equals to z, and
2. *Binding*: the sender, including PPT malicious ones, can only "control" the value of at most one of the two outputs; the value of the other output should be essentially uniformly distributed.

3.1. The NOVY commitment scheme.

Using an interactive hashing protocol as a subroutine, Naor et al. [16] constructed the following statistically hiding commitment scheme.

1. S chooses a uniform $x \leftarrow \{0,1\}^n$, and computes $z = f(x)$.
2. S and R engage in an interactive hashing protocol using z as S's private input. Let z_0 and z_1 be the common outputs, and let $z = z_d$, for some $d \in \{0,1\}$.
3. To commit to bit b, S sends $c = b \oplus d$ to R.
4. To decommit, S sends b, d, and x to R. R verifies the decommitment by checking if $c = b \oplus d$ and $z_d = f(x)$.

Let us informally argue why the above scheme constitutes a statistically hiding and computationally-binding commitment. First, we argue its hiding property. We have mentioned that z is uniform in $\{0,1\}^n$ because f is a permutation and x is chosen uniformly in $\{0,1\}^n$. By the hiding property of interactive hashing, even a computationally-unbounded malicious receiver does not know if $z = z_0$ or $z = z_1$, or equivalently, it does not know if $d = 0$ or $d = 1$. Therefore, the scheme is statistically hiding. Next, we argue its binding property. By the binding property of interactive hashing, at least one of the outputs, say z_α, is uniform in $\{0,1\}^n$ and outside the sender's control. Therefore if the sender is able to decommit to both 0 and 1, it must find a preimage of z_α. This is equivalent to finding a preimage of $f(U_n)$, and this task is computationally infeasible since f is a one-way permutation. Hence, the scheme is computationally binding.

3.2. Interactive hashing. Interactive hashing was introduced by Ostrovsky, Venkatesan, and Yung [32] in the context of oblivious transfer protocols. As mentioned above, it was applied to the construction of statistically hiding commitments by Naor et al. [16], and it will also prove to be a powerful and useful tool in our result. For our application, we will need the sender to commit to multiple bits in one execution of interactive hashing. Consequently, we extend the notion of interactive hashing to allow multiple outputs (instead of just two output strings). Since the number of outputs could be possibly superpolynomial, we succinctly describe the set of outputs as the image of a polynomial-sized circuit $C\colon \{0,1\}^k \to \{0,1\}^q$, where k and q are polynomially related to the security parameter. (We will not actually need superpolynomially many outputs in this paper, but use this more general formulation because it may be useful in future work.)

In addition to allowing for multiple outputs, our application of interaction hashing also requires a more refined notion of computational binding than the one provided by Naor, Ostrovsky, Venkatesan, and Yung [16].[3] It is for this reason we consider the notion of what it means to be a witness for a given relation W as follows: For a relation W, define the ***set of witnesses*** for z as $W_z = \{x : W(z,x) = 1\}$, and we naturally refer to any $x \in W_z$ as a ***witness*** for z. (A natural choice, utilized in the analysis of the NOVY commitment scheme described above, is $W = \{(z,x) : f(x) = z\}$ for a one-way function/permutation f.)

Definition 3.1. *An* **interactive hashing protocol with multiple outputs** *is a polynomial-time protocol $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ where both parties receive common inputs $(1^q, 1^k)$ and S_{IH} receives a private input $z \in \{0,1\}^q$. At the end of the interaction, the common output is a (polynomial-sized) circuit $C\colon \{0,1\}^k \to \{0,1\}^q$, and the private output of S_{IH} is a string $d \in \{0,1\}^k$. We call q the input length, and k the output length. The protocol $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ has to satisfy the following security properties.*

1. Correctness*: for all R^* and all $z \in \{0,1\}^q$, it is the case that $C(d) = z$, where $C = (S_{\mathrm{IH}}(z), R^*)(1^q, 1^k)$ is the common output, and $d = \mathrm{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(z), R^*)$ is the private output of S_{IH}.*[4]
2. Hiding*: for all R^*, random variables (V, D) and (V, U_k) are identically distributed, where the view of receiver R^* is $V = \mathrm{view}_{R^*}(S_{\mathrm{IH}}(U_q), R^*)$, and the*

[3] Although the notion of interactive hashing was introduced by Ostrovsky et al. [32], it was Naor et al. [16] who proved a computational binding property of interactive hashing that allows for its application to statistically hiding commitments based on any one-way permutation.

[4] The correctness property of protocols is typically defined for honest parties, in our setting this would be S_{IH} and R_{IH}. In our applications, however, it is convenient to have a stronger correctness property that also holds against malicious receivers R^*.

private output of S_{IH} *is* $D = \mathrm{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(U_q), R^*)$.

3. Binding*: there exists an oracle PPT algorithm* A *such that for every adversary* S^* *and any relation* W*, denoting the common output as* $C = (S^*, R_{\mathrm{IH}})(1^q, 1^k)$*, and private output of* S^* *as* $((x_0, d_0), (x_1, d_1)) = \mathrm{output}_{S^*}(S^*, R_{\mathrm{IH}})$*, if it is the case that*

$$\Pr[x_0 \in W_{C(d_0)} \wedge x_1 \in W_{C(d_1)} \wedge d_0 \neq d_1] > \varepsilon \ \ ,$$

where the above probability is over the coins of R_{IH} *and* S^**, then it is also the case that*

$$\Pr_{z \leftarrow \{0,1\}^q}[A^{S^*}(z, 1^q, 1^k, \varepsilon) \in W_z] > 2^{-k} \cdot (\varepsilon/q)^{O(1)} \ \ .$$

Remark 3.2. *We make three remarks regarding Definition 3.1.*

1. *The security requirements should hold for computationally unbounded* R^* *(for correctness and hiding) and computationally unbounded* S^**. In addition, the relation* W *need not be polynomial-time computable.*
2. *To simplify notation, we often write* $A^{S^*}(z)$*, or even* $A(z)$*, to denote* $A^{S^*}(z, 1^q, 1^k, \varepsilon)$*.*
3. *Although the private output of the honest sender* S_{IH} *is always a string* d*, the private output of the cheating sender* S^* *is arbitrary; hence, we can assume without loss of generality that* S^* *breaks binding by producing two pairs of strings* (x_0, d_0) *and* (x_1, d_1)*.*

The interactive hashing protocol given in [32, 16], henceforth called the NOVY Interactive Hashing, satisfies Definition 3.1 with $k = 1$. To obtain an interactive hashing protocol with multiple outputs (i.e., the case when $k > 1$), we simply end the NOVY Interactive Hashing protocol $k - 1$ rounds earlier.

Protocol 3.3. *Interactive hashing with multiple outputs* $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$*.*

Inputs:.

1. *Input length* 1^q *and output length* 1^k*, both given as common input.*
2. *String* $z \in \{0,1\}^q$*, given as private input to sender* S_{IH}*.*

Protocol:.

R_{IH}*: Select* $h_0, h_1, \ldots, h_{q-k-1}$ *such that each* h_i *is a random vector over* GF[2] *that is outside the span of* $\{h_0, h_1, \ldots, h_{i-1}\}$*.*

For $j = 0, \ldots, q-k-1$*, do the following:*

$R_{\mathrm{IH}} \to S_{\mathrm{IH}}$*: Send* h_j*.*

$S_{\mathrm{IH}} \to R_{\mathrm{IH}}$*: Send* $c_j = \langle h_j, z\rangle$*.*

Output:.

- *Common output is a circuit* $C\colon \{0,1\}^k \to \{0,1\}^q$ *computing an injective affine transformation whose image is* $\{z : \langle h_j, z\rangle = c_j \ \ \forall j = 0, \ldots, q-k-1\}$*.*
- *Private output of* S_{IH} *is a string* $d \in \{0,1\}^k$ *such that* $C(d) = z$*.*

Theorem 3.4. *There exists an interactive hashing protocol with multiple outputs, namely Protocol 3.3.*

The correctness of Protocol 3.3 is easy to see. Hence, we divide the proof of this theorem into lemmas establishing the hiding and binding properties of Protocol 3.3.

Lemma 3.5. *Protocol 3.3 satisfies the hiding property of Definition 3.1. In other words, letting interactive hashing* $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ *be as in Protocol 3.3, we have for all* R^**,* (V, D) *is distributed identically to* (V, U_k)*, where* $V = \mathrm{view}_{R^*}(S_{\mathrm{IH}}(U_q), R^*)$ *is the view of receiver* R^**, and* $D = \mathrm{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(U_q), R^*)$ *is the private output of* S_{IH}*.*

Proof. The view of any R^* will be the hash functions $h_0, h_1, \cdots, h_{q-k-1}$ together with S_{IH}'s responses $c_0, c_1, \ldots, c_{q-k-1}$. Conditioned on such a view, the sender's private input $Z = U_q$ is uniformly distributed in the k-dimensional affine subspace $\{z : \langle h_j, z\rangle = c_j \ \ \forall j\}$, and hence the value D s.t. $C(D) = Z$ is uniformly distributed in $\{0,1\}^k$. □

LEMMA 3.6. *Protocol 3.3 satisfies the binding property of Definition 3.1. That is, letting interactive hashing* $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ *be as in Protocol 3.3, there exists a oracle PPT algorithm A such that:*

> *For every* S^* *and any relation* W*, denoting the common output as* $C = (S^*, R_{\mathrm{IH}})(1^q, 1^k)$*, and private output of* S^* *as* $((x_0, d_0), (x_1, d_1)) = \mathrm{output}_{S^*}(S^*, R_{\mathrm{IH}})$*, if it is the case that*
>
> $$\Pr[x_0 \in W_{C(d_0)} \wedge x_1 \in W_{C(d_1)} \wedge d_0 \neq d_1 \in \{0,1\}^k] > \varepsilon \ \ ,$$
>
> *where the above probability is over the coin tosses of* R_{IH} *and* S^**, then it is also the case that*
>
> $$\Pr_{z \leftarrow \{0,1\}^q}[A^{S^*}(z, 1^q, 1^k, \varepsilon) \in W_z] = \Omega(\varepsilon^2 q^{-8} 2^{-k}) \ \ .$$

Proof. Note that $C(d_0)$ and $C(d_1)$ are two distinct elements in $\{0,1\}^q$ and that both elements are consistent with the transcript of the protocol, i.e, an honest S_{IH} getting each of this elements as input would have acted in the same way as S^* does in the interaction. Thus, we are in the setting of the recent interactive hashing theorem presented by Haitner and Reingold [33] and the proof follows by [33, Theorem 3.9].[5] □

3.2.1. Information-theoretic bounds.

We think of the string d as a k-bit string commitment associated to one of the 2^k outputs strings, namely $z = C(d)$, and a witness $x \in W_z = W_{C(d)}$ as a decommitment to d. Intuitively, the knowledge of x gives the sender the ability to decommit to d. The binding property, read in its contrapositive, says that if it is hard to find a witness for a uniformly random string z, then it is hard for a sender to successfully decommit to two different values. Note that this hypothesis — that it is hard to find a witness for a uniformly random z — says that for every PPT A, there is a negligible set T_A of z's on which A succeeds in finding a witness. In general, the set T_A may depend on the adversary A. In several places, however, we will only need the special case of a static set as captured in the following lemma.

LEMMA 3.7. *(Binding for Static Sets.) For any protocol* $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ *satisfying the binding condition of Definition 3.1, the following holds: For all* S^* *and any set* $T \subseteq \{0,1\}^q$*, denoting the common output as* $C = (S^*, R_{\mathrm{IH}})(1^q, 1^k)$*, we have*

$$\Pr[\exists d_0 \neq d_1 \text{ such that } C(d_0), C(d_1) \in T] < O(q^4) \cdot (\mu(T) \cdot 2^k)^{1/2}$$

where the above probability is taken over the coins of S^* *and* R_{IH} *and* $\mu(T) \stackrel{\text{def}}{=} |T|/2^q$ *denotes the density of* T.

[5] Actually, [33, Theorem 3.9] directly applies to the case that the h_i's are uniformly random vectors rather than being constrained to be linearly independent as in Protocol 3.3. However, up to q uniformly random q-dimensional vectors will be linearly independent with at least constant probability. Thus, if S^* breaks Protocol 3.3 with probability greater than ε, then it also breaks the version with uniformly random vectors with probability $\Omega(\varepsilon)$, and [33, Theorem 3.9] applies.

Setting $k = 1$ in the above lemma gives an information-theoretic bound of the NOVY Interactive Hashing; information-theoretic bounds on NOVY Interactive Hashing were studied in the context of memory-bounded oblivious transfer [34, 35, 36]. Our bound is not tight, but suffices for our applications. For tighter bounds, we refer the reader to [34, 36], or for a *constant-round* interactive hashing protocol that is binding for static sets, we refer the reader to [35].

Compare the bound of the Lemma 3.7 to the case where the adversarial sender S^* had control of only one output string. This means that the rest of the $2^k - 1$ outputs strings are distributed uniformly on $\{0,1\}^q$, and hence the bound would be roughly $\mu(T) \cdot (2^k - 1)$. The reason for this is that S^* will make the string that it controls lie in T, and the probability that at least one of the rest of the $2^k - 1$ strings lie in T is at most $\mu(T) \cdot (2^k - 1)$, by a union bound. The above bound is almost as good, and in particular if $\mu(T)$ is negligible and k logarithmic, both probabilities are negligible.

Proof. [Proof of Lemma 3.7] Define the relation $W = \{(a, b) : a \in T\}$, that is $W(a, b) = 1$ if $a \in T$ (for all values of b), and 0 if $a \notin T$ (no matter what the value of b is). Suppose there exists an S^* that with probability ε, produces two elements $d_0 \neq d_1$ such that both $C(d_0), C(d_1) \in T$. Then, by Lemma 3.6, there will be a procedure that is given a random $z \leftarrow \{0,1\}^q$ and makes $z \in T$ with probability $p = \Omega(2^{-k} \cdot \varepsilon^2/q^8)$. Since T is a fixed set, it must be the case that $p \leq \mu(T)$. This implies that $\varepsilon = O(q^4) \cdot (\mu(T) \cdot 2^k)^{1/2}$. □

4. Statistically Hiding Commitments From Known-Regular One-Way Functions. Our first hurdle is to relax the permutation structure of f to just assuming that f is a regular one-way function with *known* preimage size of say 2^{n-t}, for some known value of $t \in \{1, 2, \ldots, n\}$. This is the setting considered by Haitner et al. [25], and we review ideas from their construction in this section. To simplify the construction and analysis, we further assume f has a *known* superpolynomial security $s(n) = n^{\omega(1)}$. (Haitner et al. [25] do not make this assumption, and we will also not need it in our final construction based on an arbitrary one-way function.)

Observe that the statistical hiding property of the NOVY commitment scheme based on one-way permutation f only rely on the fact that f is a permutation because we require that $f(U_n)$ be uniform. Now if f just a regular function, then $f(U_n)$ might no longer be uniform, but instead all we can say is that $f(U_n)$ is a flat distribution with support $\mathrm{Supp}(f(U_n))$ of size 2^t. We will use *pairwise-independent hash functions*, a notion to be discussed next, to obtain an *almost-uniform* distribution from $f(U_n)$.

4.1. Hashing and randomness extraction. The ***entropy*** of a random variable X is

$$\mathrm{H}(X) = \mathop{\mathrm{E}}_{x \leftarrow X}\left[\frac{1}{\log \Pr[X = x]}\right] \quad ,$$

where here and throughout all logarithms are of base 2. This notion of entropy corresponds to *Shannon entropy* or *information entropy* in the information theory literature. Intuitively, $\mathrm{H}(X)$ measures the amount of randomness in X *on average* (in bits). For a worst-case measure of randomness, the ***min-entropy*** of X is most often used, and is defined as

$$\mathrm{H}_\infty(X) = \min_x \left[\frac{1}{\log \Pr[X = x]}\right] \quad .$$

In general $\mathrm{H}_\infty(X) \leq \mathrm{H}(X)$, but when X is flat (i.e., uniform on its support), then $\mathrm{H}(X) = \mathrm{H}_\infty(X) = \log |\mathrm{Supp}(X)|$.

A family of hash functions $\mathcal{H} = \{h : \{0,1\}^n \to \{0,1\}^m\}$ is ***pairwise independent*** (a.k.a. ***strongly 2-universal***) if for any two $x \neq x' \in \{0,1\}^n$ and any two $y, y' \in \{0,1\}^m$, when we randomly choose $h \leftarrow \mathcal{H}$, we have $\Pr[h(x) = y \text{ and } h(x') = y'] = 2^{-2m}$.

An example of a pairwise-independent family of hash functions for $n \geq m$ is the family $\mathcal{H} = \{h_{a,b} \colon \{0,1\}^n \to \{0,1\}^m\}$, where $h_{a,b}(x) = (a \cdot x + b)|_m$, arithmetic is done in the field $\mathrm{GF}(2^n)$, and $|_m$ denote taking the first m bits. We define $\ell(n,m)$ to be the number of bits required to describe an element of the hash function family $\mathcal{H}$. In our example, it takes $2n$ bits to describe each hash function $h_{a,b}$ since both a and b are elements of $\mathrm{GF}(2^n)$; hence, a family of pairwise-independent hash functions $\mathcal{H}$ mapping n-bit strings to m-bit strings exists with $\ell(n,m) = 2n$. We will use the following property of pairwise-independent hash functions to obtain an almost-uniform random variable from a random variable with sufficient min-entropy.

Lemma 4.1. *(Leftover Hash Lemma [37, 5].) Let random variable H denote a uniformly random hash function from a family of pairwise-independent hash functions $\mathcal{H}$ mapping n-bit strings to m-bit strings, and let X be a random variable taking values in $\{0,1\}^n$. For any $\varepsilon > 0$, if $\mathrm{H}_\infty(X) \geq m + 2\log(1/\varepsilon)$, and H is independent from X, then the random variable $(H, H(X))$ is ε-close in statistical distance to uniform.*

4.2. The commitment scheme. Let us return to our regular one-way function $f \colon \{0,1\}^n \to \{0,1\}^n$ with known preimage size 2^{n-t} and known security $s(n) = n^{\omega(1)}$. Consider a family of pairwise-independent hash functions $\mathcal{H} = \left\{h \colon \{0,1\}^n \to \{0,1\}^{t-\Delta}\right\}$, where $t = \mathrm{H}(f(U_n))$ and $\Delta = \frac{1}{2}\log s(n)$. Let random variable H represent a random hash function selected from $\mathcal{H}$. By the Leftover Hash Lemma 4.1, random variable $Z = (H, H(f(U_n)))$ is $(1/s(n))^{\Omega(1)}$-close to uniform, which gives statistically indistinguishability from uniform because $s(n) = n^{\omega(1)}$. So if we designate $z = (h, h(f(x)))$ as the sender's private input to the interactive hashing protocol (Protocol 3.3), even an all-powerful receiver will not get more than a negligible advantage to guess which one of the outputs is z. This hints to the following commitment scheme.

Protocol 4.2. *Commitment scheme (S, R) based on a regular one-way function $f \colon \{0,1\}^n \to \{0,1\}^n$ with known preimage size 2^{n-t} and known security $s(n) = n^{\omega(1)}$.*
Commit stage.

1. *Let $\mathcal{H} = \left\{h \colon \{0,1\}^n \to \{0,1\}^{t-\Delta}\right\}$, where $t = \mathrm{H}(f(U_n))$ and $\Delta = \frac{1}{2}\log s(n)$. S selects a uniform $x \leftarrow \{0,1\}^n$ and hash function $h \leftarrow \mathcal{H}$, and computes $y = f(x)$ and $z = (h, h(y))$.*
2. *S and R engage in interactive hashing (Protocol 3.3) with S acting as S_{IH}, R acting as R_{IH}, parameters $k = 1$ and $q = |z|$, and S_{IH} having private input z. Their common output is a circuit $C \colon \{0,1\} \to \{0,1\}^q$, and the sender receives a bit $d \in \{0,1\}$ such that $C(d) = z$.*
3. *To commit to the bit b, S sends $c = d \oplus b$ to R. The commitment of b is represented as the pair (C, c).*

Reveal stage. *To decommit, S sends bits b and d, string x, and hash function h to R. R verifies the decommitment by checking if $c = d \oplus b$ and $C(d) = (h, h(f(x)))$.*

As we have argued previously, the sender's private input z is statistically close to uniform, and hence by the hiding property of interactive hashing, this implies that the commitment scheme is statistically hiding. As for the binding property, the onewayness of f intuitively guarantees that the set T of w's for which a sender S^* can compute an element of $f^{-1}(w)$ is of density at most $1/s(n)$ in the range of f, that is the size of T is at most $2^{\mathrm{H}(f(U_n))-\log s(n)}$. Thus for any fixed h, the fraction of $z = (h, h(w))$ such that $w \in T$ is at most $2^{\mathrm{H}(f(U_n))-\log s(n)}/2^{t-\Delta} = s(n)^{-1/2} = \mathrm{neg}(n)$. By the binding property of interactive hashing stated in Lemma 3.7, the probability that S^* can force both $C(0), C(1) \in T$ is negligible and hence, the scheme is computationally binding. The complete argument to prove the binding property is actually more subtle because the set T is not actually fixed in advance, and so we need to employ the stronger binding property given in Definition 3.1. We omit the full proof, because this is merely a warm-up example to motivate our actual construction and proof in subsequent sections.

5. 1-out-of-2-Binding Commitments from Unknown-Regular One-Way Functions.

Our next hurdle is to remove to the constraint on knowing (i.e., being able to efficiently compute) the preimage size. For this setting, let us consider a regular one-way function $f\colon \{0,1\}^n \to \{0,1\}^n$ with preimage size 2^{n-t}, for an *unknown*[6] value of $t \in \{1, 2, \ldots, n\}$, but with known security $s(n) = n^{\omega(1)}$.[7] Constructing statistically hiding commitments even in this setting was still an open problem prior to our work.

Let us examine why we need to know the correct value of t in the previous scheme of Protocol 4.2. If the value of t is too high, that is $t \gg \mathrm{H}(f(U_n))$, then the scheme is no longer hiding (but would be binding). This is because the Leftover Hash Lemma 4.1 no longer applies, since in this case the min-entropy $\mathrm{H}(f(U_n))$ is too small relative to t. On the other hand, if the value of t is too low, that is $t \ll \mathrm{H}(f(U_n))$, then the scheme is no longer binding (but would be hiding). To see this, at least intuitively, observe that when t is very small, we are hashing $f(U_n)$ to a very small set $\{0,1\}^{t-\Delta}$; in other words, h collapses too many elements in $f(U_n)$. As a consequence, inverting $h(f(U_n))$ could be easy (even though inverting $f(U_n)$ is hard), and this allows us to break the binding property of our scheme.

All hope, however, is not lost. We can still use Protocol 4.2, trying all values of $t \in \{1, 2, \ldots, n\}$, to do our *first phase* commitments. And to overcome the difficulty of ensuring both hiding and binding, we will introduce a *second phase* that will be binding when $t \lesssim \mathrm{H}(f(U_n))$, and hiding when $t \gtrsim \mathrm{H}(f(U_n))$; this is obtained by the sender using a hash of the preimage x as an input to another execution of interactive hashing. This means that for the right value of $t = \mathrm{H}(f(U_n))$, both phases will be hiding, but for any value of t, at least one phase is binding. What we are describing here is a ***2-phase commitment scheme*** with a ***1-out-of-2 binding*** property, notions that we formally define in the next section.

5.1. 2-phase commitment schemes.

As mentioned previously, we will work with 2-phase commitment schemes, an alternate variant of commitments introduced by Nguyen and Vadhan [28]. These are commitment schemes with two *sequential* and *related* stages such that in each stage, the sender commits to and reveals a value.

Definition 5.1. *A* **2-phase commitment scheme** (S, R)*, with security parameter n and message lengths $(k_1(n), k_2(n))$, consists of four polynomial-time interactive protocols: the first commitment stage (S_c^1, R_c^1), the first reveal stage (S_r^1, R_r^1), the second commitment stage (S_c^2, R_c^2), and the second reveal stage (S_r^2, R_r^2). (We also write $S = (S^1, S^2)$ and $R = (R^1, R^2)$ to denote the first and second phases of S and R, respectively.) For us, both reveal phases will always be noninteractive, consisting of a*

[6] What we mean by *unknown* is that we are not able to compute the preimage size efficiently.

[7] Like in Section 4, we consider only length-preserving functions, that is $|f(x)| = |x|$ for all $x \in \{0,1\}^*$, to avoid introducing new parameters. Our construction can nevertheless be easily generalized to regular one-way functions that are not length preserving.

single message from the sender to the receiver.

1. *In the first commitment stage, S_c^1 receives a private input $\sigma^{(1)} \in \{0,1\}^{k_1}$ and coin tosses r_S. At the end of the interaction, both S_c^1 and R_c^1 output a commitment $c^{(1)}$. (Without loss of generality, we can assume that $c^{(1)}$ is the transcript of the first commitment stage.)*
2. *In the first (noninteractive) reveal stage, both S_r^1 and R_r^1 receive as common input the commitment $c^{(1)}$, and S_r^1 receives as private input its previous coin tosses r_S. S_r^1 sends R_r^1 a pair $(\sigma^{(1)}, \gamma^{(1)})$ with $\gamma^{(1)}$ interpreted as a decommitment for $\sigma^{(1)} \in \{0,1\}^{k_1}$. R_r^1 accepts or rejects based on $c^{(1)}$, $\sigma^{(1)}$, and $\gamma^{(1)}$. After that, both S_r^1 and R_r^1 output a string τ. (Without loss of generality, we can assume that τ is the transcript of the first commitment stage and the first reveal stage and includes R_r^1's decision to accept or reject.)*
3. *In the second commitment stage, both S_c^2 and R_c^2 receive as common input the string τ, and S_c^2 receives a private input $\sigma^{(2)} \in \{0,1\}^{k_2}$ and its previous coin tosses r_S. At the end of the interaction, both S_c^2 and R_c^2 output a commitment $c^{(2)}$. (Without loss of generality, we can assume that $c^{(2)}$ is the concatenation of τ and the transcript of the second commitment stage.)*
4. *In the second (noninteractive) reveal stage, both S_r^2 and R_r^2 receive as common input the commitment $c^{(2)}$, and S_r^2 receives as private input its previous coin tosses r_S. S_r^2 sends R_r^2 a pair $(\sigma^{(2)}, \gamma^{(2)})$ with $\gamma^{(2)}$ interpreted as a decommitment for $\sigma^{(2)} \in \{0,1\}^{k_2}$. R_r^2 accepts or rejects based on $c^{(2)}$, $\sigma^{(2)}$, and $\gamma^{(2)}$.*

- *We insist that scheme (S, R) have* **perfect completeness***. That is to say, if both sender S and receiver R follow their prescribed strategies, then R will always accept in both phases (with probability 1).*
- *The sender and receiver's algorithms, denoted by $S = (S^1, S^2) = ((S_c^1, S_r^1), (S_c^2, S_r^2))$ and $R = (R^1, R^2) = ((R_c^1, R_r^1), (R_c^2, R_r^2))$ respectively, are computable in polynomial time.*
- *Scheme (S, R) is* **public coin** *if all messages sent by R to S are independent random coins.*

Remark 5.2. *We make several remarks regarding Definition 5.1.*

1. *We generally consider schemes that have the same message length for both phases. When this is the case, namely $k = k_1 = k_2$, we say our 2-phase commitment scheme has message length k. It is only in Section 7 that we will use the feature of different message lengths.*
2. *Instead of providing sender S with decommitment values as private outputs of the commitment phases, we simply provide it with the same coin tosses r_S throughout (so it can recompute any private state from the transcripts of the previous phases). The receiver R, however, operates using independent coin tosses in each phase and does not need to keep private state.*

Hiding for 2-phase commitment schemes. As for standard commitment schemes, we define the security of the sender in terms of a hiding property. Stated informally, the hiding property for a 2-phase commitment scheme says that *both* commitment phases are hiding. Note that since the phases are run sequentially, the hiding property for the second commitment stage is required to hold even given the receiver's view of the first stage.

Definition 5.3. *2-phase commitment scheme (S, R), with security parameter n and message lengths $(k_1(n), k_2(n))$, is* **statistically hiding** *if for all adversarial receivers R^*,*

1. *The views of R^* when interacting with the sender in the first phase on any two messages are statistically indistinguishable. Namely, for all $\sigma^{(1)}, \widetilde{\sigma}^{(1)} \in \{0,1\}^{k_1}$, the probability ensembles* $\left\{\text{view}_{R^*}(S_c^1(\sigma^{(1)}), R^*)(1^n)\right\}_{n\in\mathbb{N}}$ *and* $\left\{\text{view}_{R^*}(S_c^1(\widetilde{\sigma}^{(1)}), R^*)(1^n)\right\}_{n\in\mathbb{N}}$ *are statistically indistinguishable.*
2. *The views of R^* when interacting with the sender in the second phase are statistically indistinguishable no matter what the sender committed to in the first phase. Namely, for all $\sigma^{(1)} \in \{0,1\}^{k_1}$, and all $\sigma^{(2)}, \widetilde{\sigma}^{(2)} \in \{0,1\}^{k_2}$, the probability ensembles* $\left\{\text{view}_{R^*}(S_c^2(\sigma^{(2)}), R^*)(\mathrm{T}, 1^n)\right\}_{n\in\mathbb{N}}$ *and* $\left\{\text{view}_{R^*}(S_c^2(\widetilde{\sigma}^{(2)}), R^*)(\mathrm{T}, 1^n)\right\}_{n\in\mathbb{N}}$, *where* $\mathrm{T} = \text{transcript}(S^1(\sigma^{(1)}), R^*)(1^n)$, *are statistically indistinguishable.*

We stress that the second condition of the above hiding definition (Definition 5.3) requires that the view of receiver in the second phase be indistinguishable for any two messages even given the transcript of the first phase, $\mathrm{T} = \text{transcript}(S^1(\sigma^{(1)}), R^*)(1^n)$.

1-out-of-2 binding for 2-phase commitment schemes. The 1-out-of-2 binding property, informally stated, says that *at least* one of the two commitment phases is binding. In other words, for every PPT malicious sender S^*, at most one of the two phases is bad in that S^* can decommit a given commitment to two different messages in that phase. We allow this bad phase to be determined dynamically by S^*. Moreover, we require that the second phase be *statistically* binding if the sender breaks the first phase. Our construction achieves this stronger property, and using it simplifies some of our proofs.

Definition 5.4. *2-phase commitment scheme (S, R), with security parameter n and message lengths $(k_1(n), k_2(n))$, is* **computationally 1-out-of-2 binding** *if there exists a "binding set" $\mathcal{B}$ of first phase transcripts such that for every function $\varepsilon(n) = 1/\operatorname{poly}(n)$, the following holds:*

1. *For all PPT adversary S^*, S^* succeeds in the following game with probability at most $\varepsilon(n)$ for all sufficiently large n:*
   (a) *S^* and R_c^1 interact and output a first-phase commitment $c^{(1)}$.*
   (b) *S^* outputs two full transcripts $\lambda = (\tau, \kappa)$ and $\widetilde{\lambda} = (\widetilde{\tau}, \widetilde{\kappa})$ of* both *phases with the following three properties:*
      - *Transcripts λ and $\widetilde{\lambda}$ both start with prefix $c^{(1)}$.*
      - *Transcript λ contains a successful opening of $c^{(1)}$ to the value $\sigma^{(1)} \in \{0,1\}^{k_1}$ using a first-phase transcript τ not in $\mathcal{B}$, and R_r^1 and R_r^2 both accept in λ.*
      - *Transcript $\widetilde{\lambda}$ contains a successful opening of $c^{(1)}$ to the value $\widetilde{\sigma}^{(1)} \in \{0,1\}^{k_1}$ using a first-phase transcript $\widetilde{\tau}$ not in $\mathcal{B}$, and R_r^1 and R_r^2 both accept in $\widetilde{\lambda}$.*

   (c) *S^* succeeds if all of the above conditions hold and $\sigma^{(1)} \neq \widetilde{\sigma}^{(1)}$.*
2. *For every (even computationally unbounded) sender S^*, the first-phase transcripts in $\mathcal{B}$ make the second phase statistically binding. In other words, for all S^*, all $\tau \in \mathcal{B}$, and all sufficiently large n, with probability at least $1 - \varepsilon(n)$ over $c^{(2)} = (S^*, R_c^2)(\tau)$, there is at most one value $\sigma^{(2)} \in \{0,1\}^{k_2}$ such that $\exists \gamma^{(2)}\ R_r^2(c^{(2)}, \sigma^{(2)}, \gamma^{(2)}) = \mathtt{accept}$.*

Some remarks on this definition:

1. Note that we require that Condition 1 to hold against PPT adversaries, but Condition 2 must hold against all, even computationally unbounded, adversaries.
2. Condition 2 has an equivalent reformulation that is more similar to Condi-

tion 1. Specifically, we can require for all S^*, all $\tau \in \mathcal{B}$, and all sufficiently large n, S^* succeeds in the following game with probability at most $\varepsilon(n)$: S^* and R_c^2 interact to get a second phase commitment $c^{(2)}$. After the interaction S^* outputs a pair of full transcripts $\lambda, \tilde{\lambda}$ and we say that S^* succeeds if both λ and $\tilde{\lambda}$ begin with prefix $c^{(2)}$ (which in turn contains τ as a prefix), R_r^2 accepts in both λ and $\tilde{\lambda}$, and λ and $\tilde{\lambda}$ contain decommitments to two distinct values $\sigma \neq \tilde{\sigma}$.

3. Conversely, an equivalent reformulation of Condition 1 says that for every PPT S^* and every $\varepsilon(n) = 1/\operatorname{poly}(n)$, the following holds for all sufficiently large n: with probability at least $1 - \varepsilon(n)$ over $c^{(1)} \overset{\text{R}}{\leftarrow} (S^*, R_c^1)$, there is at most one value $\sigma^{(1)} \in \{0,1\}^{k_1}$ for which there is at least an $\varepsilon(n)$ probability (conditioned on $c^{(1)}$) that S^* subsequently outputs a full accepting transcript $\lambda = (\tau, \kappa)$ with $\tau = (c^{(1)}, \sigma^{(1)}, \gamma^{(1)}) \notin \mathcal{B}$. Thus, we think of $c^{(1)}$ as a commitment to this unique value of $\sigma^{(1)}$.
4. Note that for S^* to break Condition 1 (or its equivalent formulation above), it must produce a full transcript that makes both R_r^1 and R_r^2 accept. Thus, the decommitment for the first phase should not really be considered "complete" until a corresponding second-phase transcript is produced. The reason we do not have the honest sender provide such a "full decommitment" during the first-phase reveal is that this would compromise the hiding property of the second phase.
5. If a 2-phase commitment is both statistically hiding and computationally 1-out-of-2 binding, then the transcripts τ generated in the first-phase interaction between the honest sender (S^1, R^1) will lie outside $\mathcal{B}$ with all but negligible probability. Indeed, the statistical hiding property of the second phase implies that the second phase cannot be statistically binding (except with negligible probability), and thus by Condition 2 the first-phase transcript τ must be outside of $\mathcal{B}$.

5.2. Our 2-phase commitment scheme. We now describe our 2-phase commitment scheme for general functions $f\colon \{0,1\}^n \to \{0,1\}^n$, not necessarily regular nor one-way—as we shall later see, it is the regularity condition that gives the hiding property, and the one-wayness of the function that gives the binding property of our scheme. Let $\mathcal{H} = \{h\colon \{0,1\}^n \to \{0,1\}^m\}$ be a family of pairwise-independent hash functions. As shown in Section 4.1, we have a family whose description of each element takes $\ell(n,m) = 2n$ bits. It will be convenient to make $\ell(n,m) + m = q(n)$, for some fixed polynomial $q(n)$, so that for every $y \in \{0,1\}^n$, $|h, h(y)| = q(n)$. This can be done by padding random bits to the description of h.

In addition, it will be convenient to work with protocols where the sender has no input $\sigma^{(j)}$ to be committed to, but rather privately receives an output $d^{(j)}$ at the end of each phase $j \in \{1,2\}$ of the commitment. If we can ensure that $d^{(j)}$ is close to uniform given the receiver's view, such a protocol can be easily tuned into a commitment scheme: the sender can commit to an $\sigma^{(j)}$ of its choice by sending $d^{(j)} \oplus \sigma^{(j)}$ at the end of the commit stage.

On the other hand, it will be convenient to have a special private input $x \in \{0,1\}^n$ that corresponds to a portion of sender's coin tosses. The reason for this is that later we will want to discuss properties of the protocol when x is chosen randomly from a subset $\Gamma \subseteq \{0,1\}^n$ rather than uniformly from the entire space.

Protocol 5.5. *2-phase commitment scheme (S,R) based on $f\colon \{0,1\}^n \to \{0,1\}^n$.*

Parameters: *Integers* $t \in \{1, 2, \ldots, n\}$, $k_1 = k_2 = k \in \{1, 2, \ldots, n\}$, $\Delta_1 \in \{0, 1, \ldots, t\}$, *and* $\Delta_2 \in \{0, 1, \ldots, n - t\}$.

Sender's (special) private input: *String* $x \in \{0,1\}^n$. *(Again, this is not the value to which the sender is committing, but is rather part of its coins, which will be chosen uniformly at random by* S *unless otherwise specified.)*

First phase commit:

1. S_c^1 *sets* $y = f(x)$.
2. *Let* $\mathcal{H}_1 = \{h_1 \colon \{0,1\}^n \to \{0,1\}^{t-\Delta_1}\}$ *be a family of pairwise-independent hash functions.* S_c^1 *chooses a random hash* $h_1 \leftarrow \mathcal{H}_1$, *and computes* $z^{(1)} = (h_1, h_1(y)) \in \{0,1\}^q$.
3. (S_c^1, R_c^1) *run the interactive hashing protocol* $(S_{\mathrm{IH}}(z^{(1)}), R_{\mathrm{IH}})(1^q, 1^k)$, *given by Protocol 3.3, with* S_c^1 *and* R_c^1 *acting as* S_{IH} *and* R_{IH} *respectively.*
   Let circuit $C^{(1)} \colon \{0,1\}^k \to \{0,1\}^q$ *be the common output and* $d^{(1)} \in \{0,1\}^k$ *(with* $C^{(1)}(d^{(1)}) = z^{(1)}$*) be* S_{IH}*'s private output in*
   $(S_{\mathrm{IH}}(z^{(1)}), R_{\mathrm{IH}})(1^q, 1^k)$.

First phase sender's private output: String $d^{(1)} \in \{0,1\}^k$.

First phase reveal:

S_r^1 *sends the tuple* $\gamma^{(1)} = (d^{(1)}, y, h_1)$.
Receiver R_r^1 *accepts if and only if* $C^{(1)}(d^{(1)}) = (h_1, h_1(y))$.

Second phase commit:

Second phase common input: First-phase transcript $\tau = \mathrm{transcript}(S^1(x), R^1)$, *which in particular includes the string* y.

1. *Let* $\mathcal{H}_2 = \{h_2 \colon \{0,1\}^n \to \{0,1\}^{n-t-\Delta_2}\}$ *be a family of pairwise-independent hash functions.* S_c^2 *chooses a random hash* $h_2 \leftarrow \mathcal{H}_2$, *and computes* $z^{(2)} = (h_2, h_2(x)) \in \{0,1\}^q$.
2. (S_c^2, R_c^2) *run the interactive hashing protocol* $(S_{\mathrm{IH}}(w), R_{\mathrm{IH}})(1^q, 1^k)$, *given by Protocol 3.3, with* S_c^2 *and* R_c^2 *acting as* S_{IH} *and* R_{IH} *respectively.*
   Let circuit $C^{(2)} \colon \{0,1\}^k \to \{0,1\}^q$ *be the common output and* $d^{(2)} \in \{0,1\}^k$ *be* S_{IH}*'s private output in* $(S_{\mathrm{IH}}(z^{(2)}), R_{\mathrm{IH}})(1^q, 1^k)$.

Second phase sender's private output: String $d^{(2)} \in \{0,1\}^k$.

Second phase reveal:

S_r^2 *sends the tuple* $\gamma^{(2)} = (d^{(2)}, x, h_2)$.
Receiver R_r^2 *accepts if and only if* $f(x) = y$ *and* $C^{(2)}(d^{(2)}) = (h_2, h_2(x))$.

THEOREM 5.6. *Suppose* $f : \{0,1\}^n \to \{0,1\}^n$ *is a regular one-way function with (known) security* $s(n) = n^{\omega(1)}$. *Then for any setting of parameters such that* $t \in [\mathrm{H}(f(U_n)), \mathrm{H}(f(U_n)) + 1)$, $k = O(\log n)$, *and* $\Delta_1 = \Delta_2 = \frac{1}{4}\log s \le \max\{t, n - t\}$, *Protocol 5.5 is a 2-phase commitment scheme that is statistically hiding and computationally 1-out-of-2 binding. Moreover the computational 1-out-of-2 binding property holds regardless of the setting of* t *or the regularity of* f.

In order to apply the above theorem, it needs to be verified that if we set $t = \lfloor \mathrm{H}(f(U_n)) \rfloor$, then $\frac{1}{4}\log s \le \min\{t, n-t\}$. This can be easily made to hold by padding the input of f. Specifically, given a one-way function $f : \{0,1\}^n \to \{0,1\}^n$, the function $f' : \{0,1\}^{3n} \to \{0,1\}^{3n}$ defined by $f'(x, y, z) = (f(x), y)$ for $|x| = |y| = |z|$ is a regular one-way function with security $s \le 2^n$ and output entropy $\mathrm{H}(f'(U_{n'})) \in [n, 2n]$, so $\frac{1}{4}\log s \le n/4 < \min\{t, n' - t\}$, where $n' = 2n$ is the input length of f'.

Because we do not know how to efficiently compute the correct value of $t = \mathrm{H}(f(U_n))$, we are forced to try out all values of $t = 1, 2, \ldots, n$ to get a collection of commitment schemes, as stated in the next corollary. While having a collection of schemes instead of a single scheme may seem inconvenient, in Section 7 we will show how to convert such a collection of 2-phase commitments into a *single* commitment scheme that is statistically hiding and computationally binding (in the standard sense of binding).

Corollary 5.7. *Given a regular one-way function $f : \{0,1\}^n \to \{0,1\}^n$ with known security $s(n) = n^{\omega(1)}$, we can construct in time polynomial in n a collection of public-coin 2-phase commitment schemes $\mathcal{COM} = \{\mathsf{Com}_1, \cdots, \mathsf{Com}_n\}$, such that:*

- *there exists an index $i \in \{1, 2, \ldots, n\}$ such that scheme Com_i is statistically hiding, and*
- *for every index $i \in \{1, 2, \ldots, n\}$, scheme Com_i is computationally 1-out-of-2 binding.*

For notational convenience and generality, the above corollary and some of our subsequent results are stated in terms of finite functions $f : \{0,1\}^n \to \{0,1\}^n$ for a fixed value n of the security parameter. When we say that the function f is 'given', this can be interpreted as being given the boolean circuit computing f (or, more generally, given f as an oracle), and 'constructing' the commitment schemes Com_i can be interpreted as constructing the boolean circuits (or, more generally, circuits with oracle gates for evaluating f) that compute the next-message functions of the protocol.

We divide the proof of Theorem 5.6 into Lemma 5.8 and Lemma 5.9 that establish the statistical hiding and computational 1-out-of-2 binding properties of Protocol 5.5, respectively.

Lemma 5.8. *If $f : \{0,1\}^n \to \{0,1\}^n$ is a regular function, then Protocol 5.5, with any setting of parameters satisfying $t \in [\mathrm{H}(f(U_n)), \mathrm{H}(f(U_n)) + 1)$, $k < q(n)$, $\Delta_1 = \Delta_2 = \omega(\log n) \leq \min\{t, n - t\}$, is statistically hiding in the sense of Definition 5.3.*

Proof. Since $t \leq \mathrm{H}(f(U_n)) + 1$, the Leftover Hash Lemma (Lemma 4.1) tells us that random variable $Z^{(1)} = (H_1, H_1(f(U_n)))$ is $2^{-\Omega(\Delta_1)}$-close to the uniform. Then by the hiding property of interactive hashing (Definition 3.1), the first commitment phase is $2^{-\Omega(\Delta_1)}$-hiding, which suffices because $\Delta_1 = \omega(\log n)$.

Let τ be the transcript of the first phase and y the string sent in the first reveal phase. Let random variable X represent selecting at random a string from the set $f^{-1}(y)$. Since X is a flat source with entropy $n - \mathrm{H}(f(U_n)) \geq n - t$, and h_2 maps to strings of length $n - t - \Delta_2$, we apply the Leftover Hash Lemma once more to conclude that random variable $Z^{(2)} = (H_2, H_2(X))$ is $2^{-\Omega(\Delta_2)}$-close to the uniform, even given τ. By the hiding property of interactive hashing, the second commitment phase is $2^{-\Omega(\Delta_2)}$-hiding, which in turn is statistically hiding since $\Delta_2 = \omega(\log n)$. □

Lemma 5.9. *If $f : \{0,1\}^n \to \{0,1\}^n$ is a $s(n)$-secure one-way function (not necessarily regular), then for any value of $t \in \{1, 2, \ldots, n\}$, Protocol 5.5, with any setting of parameters such that $k = O(\log n)$, $\Delta_1 = \Delta_2 \leq \min\{\frac{1}{4}\log(s(n)), t, n - t\}$, is computationally 1-out-of-2 binding in the sense of Definition 5.4.*

Intuitively, the proof of Lemma 5.9 will proceed as follows. We will show that w.h.p. the first commit phase binds the (cheating) sender to a unique "heavy" string y, ie a string whose probability mass under $f(U_n)$ is at least roughly 2^{-t}. During the first reveal phase, the sender can choose to reveal either this heavy string or a "light" string. In the former case, the first phase is computationally binding, and in the latter, the second phase is statistically binding.

Formally, we define the binding set $\mathcal{B}$ as follows:

> For every $t \in \{1, 2, \ldots, n\}$, define the set of *light* strings to be $L_t = \{y \in \{0,1\}^n : \Pr[f(U_n) = y] \leq 2^{-t-\Delta_3}\}$, for a parameter Δ_3 that we

will set at the end of the proof. Define the binding set $\mathcal{B}$ to be the set of transcripts where the sender reveals $y \in L_t$.

Now, we break the proof of Lemma 5.9 into Claims 5.10 and 5.11 below, which establish the binding property for the first and second phases, respectively.

Claim 5.10. *For the binding set $\mathcal{B}$ defined above, if there exists a PPT S^* that succeeds with probability $\varepsilon = \varepsilon(n)$ in the game in Condition 1 of Definition 5.4, then there exists a PPT B that can invert f with success probability at least*

$$\varepsilon^{O(1)} \cdot 1/\operatorname{poly}(n) \cdot 2^{-(k+\Delta_1+\Delta_3)} \quad .$$

Proof. We define a relation W as follows:

$$W = \left\{ (z^{(1)}, x) : \exists h_1 \text{ such that both } z^{(1)} = (h_1, h_1(f(x))) \text{ and } f(x) \notin L_t \right\} \quad .$$

Suppose we have a PPT S^* that succeeds with probability greater than ε in the game described in Condition 1 of Definition 5.4. In particular, this means that S^* after interacting with R_{IH} will, with probability greater than ε, produce pairs $(d_0^{(1)}, x_0)$ and $(d_1^{(1)}, x_1)$ such that $d_0^{(1)} \neq d_1^{(1)}$, $(C^{(1)}(d_0), x_0) \in W$, and $(C^{(1)}(d_1), x_1) \in W$. By the binding property of interactive hashing (Condition 3 of Definition 3.1), there exists a PPT A such that

$$\Pr_{z^{(1)} \leftarrow \{0,1\}^q}[A(z^{(1)}) \in W_{z^{(1)}}] > 2^{-k} \cdot \left(\frac{\varepsilon}{q}\right)^{O(1)} \quad , \tag{5.1}$$

where the above probability is taken over the coin tosses of A and $v \leftarrow \{0,1\}^q$.

Consider an algorithm B that on input y, picks a random hash function $h_1 \leftarrow \mathcal{H}_1$, and outputs $A(h_1, h_1(y))$. We let $\eta = h_1(y)$, and compute the probability that B inverts f as follows:

$$\begin{aligned}
&\Pr[B(f(U_n)) \in f^{-1}(f(U_n))] \\
&\quad = \mathop{\mathrm{E}}_{h_1 \leftarrow \mathcal{H}_1}\left[\sum_x \Pr[A(h_1, h_1(f(U_n))) = x \wedge f(x) = f(U_n)]\right] \\
&\quad = \mathop{\mathrm{E}}_{h_1 \leftarrow \mathcal{H}_1}\left[\sum_{\eta, x \text{ s.t. } \eta = h_1(f(x))} \Pr[f(U_n) = f(x)] \cdot \Pr[A(h_1, \eta) = x]\right] \\
&\quad \geq \mathop{\mathrm{E}}_{h_1 \leftarrow \mathcal{H}_1}\left[\sum_{\eta, x \text{ s.t. } x \in W_{(h_1,\eta)}} \Pr[f(U_n) = f(x)] \cdot \Pr[A(h_1, \eta) = x]\right] \\
&\quad \geq \mathop{\mathrm{E}}_{h_1 \leftarrow \mathcal{H}_1}\left[\sum_{\eta, x \text{ s.t. } x \in W_{(h_1,\eta)}} 2^{-t-\Delta_3} \cdot \Pr[A(h_1, \eta) = x]\right] && (x \in W_{(h_1,\eta)} \Rightarrow f(x) \notin L_t) \\
&\quad = 2^{-t-\Delta_3} \cdot 2^{t-\Delta_1} \cdot \Pr_{(h_1,\eta) \leftarrow \mathcal{H}_1 \times \{0,1\}^{t-\Delta_1}}[A(h_1, \eta) \in W_{(h_1,\eta)}] \\
&\quad > 2^{-(\Delta_1+\Delta_3)} \cdot 2^{-k} \cdot \left(\frac{\varepsilon}{q}\right)^{O(1)} && (\text{by (5.1)}) \\
&\quad = \varepsilon^{O(1)} \cdot 1/\operatorname{poly}(n) \cdot 2^{-(k+\Delta_1+\Delta_3)} && (\text{since } q = \operatorname{poly}(n)) \quad .
\end{aligned}$$

□

Claim 5.11. *For the binding set $\mathcal{B}$ defined above, Condition 2 of Definition 5.4 is satisfied with $\varepsilon = \mathrm{poly}(n) \cdot 2^{-\Omega(\Delta_3 - \Delta_2)} \cdot 2^{k/2}$.*

Proof. Let $y \in L_t$ be the string sent in the first reveal phase. This means that $\Pr[f(U_n) = y] \leq 2^{-t-\Delta_3}$, or equivalently $\left|f^{-1}(y)\right| \leq 2^{n-t-\Delta_3}$. Define set $T = \{(h_2, h_2(x)) : h_2 \in \mathcal{H}_2, x \in f^{-1}(y)\}$, and let $\mu(T)$ denote the density of the subset T. Since h_2 maps $\{0,1\}^n$ to $\{0,1\}^{n-t-\Delta_2}$, we have

$$\mu(T) \leq \frac{\left|f^{-1}(y)\right|}{2^{n-t-\Delta_2}} \leq \frac{2^{n-t-\Delta_3}}{2^{n-t-\Delta_2}} = 2^{\Delta_2 - \Delta_3} \ .$$

Applying Lemma 3.7, we have

$$\Pr\left[(z_0^{(2)}, z_1^{(2)}) = \mathrm{output}(S^*, R_{\mathrm{IH}}) \text{ satisfies } \ z_0^{(2)}, z_1^{(2)} \in T\right] < \mathrm{poly}(q) \cdot 2^{-\Omega(\Delta_3 - \Delta_2)} \cdot 2^{k/2} \ ,$$

which then concludes our proof since q is a fixed polynomial in n. □

Proof. [Proof of Lemma 5.9] Set $\Delta_3 = \frac{1}{2} \log s(n)$, and we are given that $k = O(\log n)$, and $\Delta_1 = \Delta_2 \leq \frac{1}{4} \log(s(n))$. With this setting, Claim 5.11 shows that Condition 2 in Definition 5.4 is satisfied with $\varepsilon(n) = \mathrm{poly}(n) \cdot 2^{-\Omega(\log s(n))} = \mathrm{neg}(n)$, since $s(n) = n^{\omega(1)}$. Condition 1 of Definition 5.4 is also satisfied with negligible probability $\varepsilon(n)$ because otherwise f can be inverted with probability

$$\begin{aligned} \varepsilon^{O(1)} \cdot 1/\mathrm{poly}(n) \cdot 2^{-(k+\Delta_1+\Delta_3)} &\geq \varepsilon^{O(1)} \cdot 1/\mathrm{poly}(n) \cdot 2^{-(O(\log n) + (3/4)\cdot(\log s(n)))} \\ &= \varepsilon^{O(1)} \cdot 1/\mathrm{poly}(n) \cdot s(n)^{-3/4} \ , \end{aligned}$$

which is greater than $1/s(n)$ if ε is nonnegligible. □

6. 1-out-of-2-Binding Commitments From Any One-Way Function. Our next hurdle is to remove the regularity assumption. It turns out that this is the most technically challenging step. (We also need to remove the need for known security, but this will follow as a byproduct of our approach to removing the regularity assumption.) Similar to our construction from regular one-way functions (with unknown preimage size) in Section 5, our construction based on any one-way function yields a collection two-phase commitments, as stated below.

Theorem 6.1. *Given a one-way function $f : \{0,1\}^n \to \{0,1\}^n$, we can construct in time polynomial in n a collection of $m = \mathrm{poly}(n)$ public-coin 2-phase commitment schemes $\mathcal{COM} = \{\mathsf{Com}_1, \cdots, \mathsf{Com}_m\}$ with message lengths $(k_1, k_2) = (n, n)$, such that:*

- *there exists an index $i \in \{1, 2, \ldots, m\}$ such that scheme Com_i is statistically hiding, and*
- *for every index $i \in \{1, 2, \ldots, m\}$, scheme Com_i is computationally 1-out-of-2 binding.*

Note that this theorem provides 2-phase commitment schemes for long messages, specifically ones with length equal to the input length n of the one-way function f. Essentially the same proof can provide schemes with message length $k(n)$ for any desired polynomial k. Alternatively, we can apply the above theorem to the function $f'(x_1, \ldots, x_k) = (f(x_1), \ldots, f(x_k))$, which has input length $n' = k \cdot n$ and is one-way if f is.

A collection of two-phase commitment schemes as above turns out to suffice for obtaining statistical zero-knowledge arguments for all of NP (see [28, 1]). Hence, Theorem 6.1 suffices to establish Theorem 1.2, which states that statistical zero-knowledge arguments for all of NP can be based on any one-way function. However, in Section 7, we will show how to transform the above collection of two-phase commitment schemes and into a *single* commitment scheme that is statistically hiding and computationally binding (in the standard sense of binding). This proves Theorem 1.1, the main theorem of this paper, and and gives a more modular proof of Theorem 1.2 (simply by plugging the commitments into [9]).

We prove Theorem 6.1 in Sections 6.1 through 6.3.

6.1. Overview. We now present an overview of how we generalize our construction for regular one-way functions with unknown preimage size (Protocol 5.5) to arbitrary one-way functions. As shown in Lemma 5.9, this protocol already achieves 1-out-of-2 binding when f is any one-way function (for every value of t). Thus the challenge is the hiding property. (Another issue is that Protocol 5.5 requires a one-way function with known security. It turns out that our method for handling the hiding property also eliminates the need to know the security.)

As discussed in Section 5, for regular one-way functions with unknown preimage size, Protocol 5.5 has a hiding first phase when the parameter t satisfies $t \lesssim \mathrm{H}(f(U_n))$ and has a hiding second phase when t satisfies $t \gtrsim \mathrm{H}(f(U_n))$. Intuitively, when f is not regular, we should replace the fixed value $\mathrm{H}(f(U_n))$ with the dynamic value $\mathrm{H}_f(y) \stackrel{\text{def}}{=} \log(1/\Pr[f(U_n) = y])$, where $y = f(x)$ is the value chosen by the sender in the pre-processing step, because $\mathrm{H}_f(y)$ can be viewed as measuring the amount of *entropy* in y. The *approximable preimage-size one-way functions* studied by Haitner et al. [25] come equipped with an algorithm that estimates $\mathrm{H}_f(y)$, but for general one-way functions, this quantity may be infeasible to compute.

A weakly hiding scheme (details in Section 6.2). One natural approach is to have the sender choose t at random and hope that it is close to $\mathrm{H}_f(y)$. When we choose t too small, only the first phase will be hiding, and when we choose t too large, only the second phase will be hiding. But we have a non-negligible probability of $\delta = 1/n$ that $t \approx \mathrm{H}_f(y)$, and thus both phases will be hiding. Thus we have a 1-out-of-2-binding commitment scheme satisfying a ***weak hiding*** property, where with probability $\delta = 1/n$, both phases are hiding, and it is always the case that at least one phase is hiding. Actually, in order to simplify our analysis, we will include t as a parameter to the protocol. Then there exists a choice of t such that the protocol is weakly hiding in the sense above, and for all choices of t the protocol is 1-out-of-2 binding. At the end, we will enumerate over all values of t, resulting in a *collection* of commitment schemes as claimed in Theorem 6.1, albeit with a weak hiding property.

Unfortunately, we do not know how to directly construct statistically hiding commitments from weakly hiding 1-out-of-2-binding commitments. Thus instead, much of the effort in this paper is devoted to amplifying the weak hiding property, where $\delta = 1/n$, into a ***strong hiding*** property, where $\delta = 1 - \mathrm{neg}(n)$, while maintaining the 1-out-of-2 binding property.[8]

Amplifying the hiding property (details in Section 6.3). We do not amplify the hiding probability from $\delta = 1/n$ to $\delta = 1 - \mathrm{neg}(n)$ in one shot, but instead perform a sequence of $\log n$ iterations, each one of which increases δ by roughly a factor of 2.

[8] One may wonder why we do not now apply the compiler of Section 7, which converts strongly hiding 2-phase commitments into standard commitments. Even if that compiler could be made to work for weakly hiding 2-phase commitments (producing a weakly hiding standard commitment), another obstacle is that the compiler requires a 2-phase commitment with polynomial message length, whereas our weakly hiding commitment only has a logarithmic message length (similarly to Theorem 5.6). It turns out that our amplification procedure also increases the message length as needed.

This results in $\delta = \Omega(1)$, and then we are able to get $\delta = 1 - \text{neg}(n)$ in just one more amplification step.

How do we double δ? A natural idea is to consider several executions of the previous weakly hiding scheme. Specifically, if we take $m = O(1)$ executions, the probability that at least one of the executions has both phases hiding is roughly $m \cdot \delta$. Moreover, each of the remaining $m - 1$ executions have either the first phase hiding or the second phase hiding. Thus for some value of β, there are $\beta + 1$ first phases that are hiding and $m - \beta$ second phases that are hiding. It turns out that we can choose β so that this exact $(\beta + 1, m - \beta)$ breakdown given that one execution has both phases hiding occurs with probability $\Omega(1/\sqrt{m})$. Thus we are in the situation described with probability $m \cdot \delta \cdot \Omega(1/\sqrt{m}) = \Omega(\sqrt{m} \cdot \delta) > 2\delta$, for a large enough constant m.

Now our aim is to combine the outcomes of the weakly hiding schemes in such a way that when the above-described situation occurs, which happens with probability at least 2δ, both phases are hiding. Notice that the secret values $\sigma_1, \ldots, \sigma_m \in \{0,1\}^k$ to which the sender commits in the first commit phases have entropy (even min-entropy) at least $(\beta + 1) \cdot k$ conditioned on the receiver's view (because $(\beta + 1)$ of them are hiding), and similarly the sender's secrets in the second commit phases have entropy at least $(m - \beta) \cdot k$ conditioned on the receiver's view. Let us compare this to the situation with binding. Since each execution is 1-out-of-2 binding, a cheating polynomial-time sender can break the binding property for either at most β of the first phases *or* at most $m - \beta - 1$ of the second phases. Thus the number of possible values to which the sender can open in each case is at most $2^m \cdot 2^{k \cdot \beta}$ in the first phase *or* at most $2^{k \cdot (m - \beta - 1)}$, where the 2^m factor in the first bound comes from the sender's ability to choose which subset of executions to break (and it is this factor that limits us to taking m to be a constant). We can view these as strong forms of entropy upper bounds $m + k\beta$ and $k \cdot (m - \beta - 1)$. In at least one phase, there will be an *entropy gap* of at least $k - m$.

How can we exploit these entropy gaps? It turns out that interactive hashing, again, is a useful tool. Specifically, in the first phase we have the sender choose a random pairwise-independent hash function h_1 mapping to approximately $(\beta + 1) \cdot k$ bits and use $(h_1, h_1(\sigma_1, \ldots, \sigma_m))$ as the input to the interactive hashing protocol, and analogously for the second phase. By the Leftover Hash Lemma, this pairwise-independent hashing results in an almost-uniform distribution, so the interactive hashing hiding property applies. As for the binding property, the bound on the number of the sender's choices gets translated to saying that the sender's input (in the first phase) comes from a set Γ of density $2^{-(k-m)}$, so the interactive hashing binding property applies. The analysis for the second phase are similar. Formalizing these ideas, we get a new 1-out-of-2-binding commitment scheme in which both phases are hiding with probability at least 2δ.

When we try to iterate this amplification step $O(\log n)$ times, we run into a new difficulty. Specifically, the above sketch hides the fact that we pay entropy losses of $\omega(\log n)$ in both the hiding and binding analysis. The entropy loss of $\omega(\log n)$ in the hiding property comes from the Leftover Hash Lemma, in order to ensure that $(h_1, h_1(\sigma_1, \ldots, \sigma_m))$ has negligible statistical distance from uniform. The entropy loss of $\omega(\log n)$ in the binding property comes from needing the $\mu(\Gamma) \cdot 2^k$ factor to be negligible when applying Lemma 3.7. This forces us to go, in one step of amplification, from a commitment scheme for secrets of length k to a scheme for secrets of length $k - m - \omega(\log n)$. As in Lemma 5.9, we can take the initial secret to be of length $k = \Theta(\log s(n)) = \omega(\log(n))$ if the one-way function has known security $s(n) = n^{\omega(1)}$.

But to tolerate $\log n$ losses of $\omega(\log n)$, we would need $s(n) = n^{\omega(\log n)}$ (i.e., at least quasi-polynomial security).

To get around this difficulty, in the amplification, we work with more relaxed, average-case measures of entropy for both the hiding and binding analysis. Specifically, for hiding, we keep track of the expected collision probability of the sender's secret, conditioned on the receiver's view. (Actually, we use the expected square root of the collision probability.) For binding, we work with the expected number of values to which the sender can open. In both cases, we only require these expectations to be within a constant factor of the ideal values, which are 2^{-k} and 1 respectively. With these measures, it turns out that we need only lose $O(m) = O(1)$ bits in the entropy gap with each amplification step. Moreover, once we amplify δ to a constant, we can afford to take the number of executions m to equal the security parameter n and get an $\Omega(n)$-bit entropy gap in the final amplification step. This allows us to achieve exponentially strong statistical hiding even when we do not know the security and start with secret length of $k = O(\log n)$.

The hiding analysis of the above construction works only for certain values of t in the weakly hiding scheme, and for certain values of the β's in the amplification steps. We try out all possible values of t and β's, thus obtaining a collection of $\mathrm{poly}(n)$ schemes, at least one of which is strongly hiding and all of which are 1-out-of-2 binding. Notice that the number of possible choices of t and the β's are polynomial in n since $t \in \{1, 2, \ldots, n\}$, the β's in the each step except for the last is in the range $\{0, 1, \ldots, m-1\}$, for some constant m, and the last β is in the range $\{0, 1, \ldots, n\}$.

6.2. Weakly hiding and 1-out-of-2-binding commitments. As discussed in Section 5, for the case of regular one-way functions with unknown preimage size, Protocol 5.5 has a hiding first phase when the parameter t satisfies $t \lessapprox \mathrm{H}(f(U_n))$ and has a hiding second phase when t satisfies $t \gtrapprox \mathrm{H}(f(U_n))$. When f is not regular, then there will be one value of $t \in \{1, 2, \ldots, n\}$ such that $\mathrm{H}_f(y) \approx t$ with probability $\gtrsim 1/n$ over $y \overset{\mathrm{R}}{\leftarrow} f(U_n)$, where $\mathrm{H}_f(y) \overset{\mathrm{def}}{=} \log(1/\Pr[f(U_n) = y])$ measures the entropy in the specific output y). This is the case because there are only n possible choices for the value of t.

With this observation in mind, our two-phase commitment scheme from general one-way functions will be the same as the scheme in Protocol 5.5, with setting of parameters $t = t_0$, $k = O(\log n)$, and $\Delta_1 = \Delta_2 = 2 \log n$, for some $t_0 \in \{1, 2, \ldots, n\}$. In other words, the same scheme—with slightly different setting of parameters—used in the case of regular one-way functions is also applicable to general one-way functions.

This commitment scheme (using general one-way functions), as we will show, is computationally 1-out-of-2 binding, but only statistically hiding in both phases with probability at least $1/n$ (hence, called ***weakly hiding***). In order to obtain a tighter analysis when we amplify this scheme, we depart from the standard measures of hiding and binding used in Section 5. Instead, we measure the statistical hiding property of our two-phase commitments using the *expected square root of the collision probability* of the sender's secret, denoted as $\mathrm{CP}^{1/2}$, and defined in Section 6.2.1. We measure the binding property by analyzing the *expected number of values* to which an adversarial sender can open.

Later in Section 6.3, we show how to boost the statistical hiding probability to $1 - 2^{-\Omega(n)}$ while maintaining the computational 1-out-of-2 binding property.

6.2.1. Properties of collision probability.

Definition 6.2. *For any random variable A, we define its* **collision probability** *as the probability that two independent samples from A are equal. In other words,*

$$\mathrm{CP}(A) \stackrel{\mathrm{def}}{=} \sum_{a \in \mathrm{Supp}(A)} (\Pr[A = a])^2 = \mathop{\mathrm{E}}_{a \leftarrow A} [\Pr[A = a]] \ .$$

Measuring the collision probability of a random variable is equivalent to measuring its ***Renyi entropy of order 2***, defined as

$$\mathrm{H}_2(A) = \log \frac{1}{\mathrm{E}_{a \leftarrow A} [\Pr[A = a]]} = \log \frac{1}{\mathrm{CP}(A)} \ .$$

Definition 6.3. *For any random variable A, we denote the square root of its collision probability as:*

$$\mathrm{CP}^{1/2}(A) \stackrel{\mathrm{def}}{=} \sqrt{\mathrm{CP}(A)} \ .$$

For any two (possibly correlated) random variables A and B, we define the **expected square root of A's collision probability given** *B as*

$$\mathrm{CP}^{1/2}(A|B) \stackrel{\mathrm{def}}{=} \mathop{\mathrm{E}}_{b \leftarrow B} \left[\mathrm{CP}^{1/2}(A|_{B=b}) \right] \ .$$

We think of $\mathrm{CP}^{1/2}(A|B) \leq \sqrt{2^k}$ as saying that A has ***conditional Renyi entropy*** of at least k given B. We use the expected *square root* of the collision probability (as our measure of hiding) instead of just expected collision probability in order to ensure that conditioning on a random variable Z can only decrease the conditional Renyi entropy by at most $\log(|\mathrm{Supp}(Z)|)$ bits. (See Lemma 6.7 below for details.)

The following lemmas show that $\mathrm{CP}^{1/2}$ behaves nicely as an entropy measure. Proofs are in Appendix A.

Lemma 6.4. *For independent pairs of random variables $(X_1, Y_1), \ldots, (X_m, Y_m)$,*

$$\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) = \prod_{i=1}^{m} \mathrm{CP}^{1/2}(X_i|Y_i) \ .$$

Note that X_i and Y_i can be correlated, it is only required that the pair (X_i, Y_i) be independent from the other tuples.

In terms of conditional Renyi entropy, Lemma 6.4 states that the entropy is additive for independent random variables. We will actually need a generalization of Lemma 6.4 to deal with somewhat dependent random variables, as stated in the next lemma.

Lemma 6.5. *Suppose random variables $(X_1, Y_1), \ldots, (X_m, Y_m)$ satisfy the following conditions for some values of $\alpha_1, \ldots, \alpha_m \in \mathbb{R}^+$ and all $i = 1, 2, \ldots, m$:*

1. *For every $(y_1, \ldots, y_{i-1}) \in \mathrm{Supp}(Y_1, Y_2, \ldots, Y_{i-1})$,*

$$\mathrm{CP}^{1/2}(X_i|_{Y_1=y_1,\ldots,Y_{i-1}=y_{i-1}} \mid Y_i|_{Y_1=y_1,\ldots,Y_{i-1}=y_{i-1}}) \leq \alpha_i \ .$$

2. *For every $(y_1, \ldots, y_i) \in \mathrm{Supp}(Y_1, Y_2, \ldots, Y_i)$, the $i + 1$ random variables $X_1, X_2, \ldots, X_i$, and Y_{i+1} are independent after conditioning on $Y_1 = y_1, \ldots, Y_i = y_i$.*

Then,

$$\mathrm{CP}^{1/2}((X_1,\ldots,X_m)|(Y_1,\ldots,Y_m)) \le \prod_{i=1}^{m} \alpha_i \ .$$

The next lemma shows that pairwise-independent randomness extraction $(h, h(x))$ preserves the $\mathrm{CP}^{1/2}$ measure.

Lemma 6.6. *(Randomness Extraction Lemma.) Let (X,Y) be any (possibly correlated) pair of random variables, and let random variable H denote a random hash function from a family of pairwise-independent hash functions $\mathcal{H}$ with range $\{0,1\}^\alpha$. Suppose the hash functions from $\mathcal{H}$ are represented by $(q-\alpha)$-bit strings and $\mathrm{CP}^{1/2}(X|Y) \le \sqrt{2^{-(\alpha+3)}}$. If H is independent from (X,Y), then*

$$\mathrm{CP}^{1/2}((H,H(X))|Y) \le \sqrt{2^{-(q-1)}} \ .$$

In other words, if X has at least $\alpha+3$ bits of conditional Renyi entropy given Y, then we can extract α bits from X that have conditional Renyi entropy at least $\alpha-1$. Notice that this entropy loss is only 4 bits, as compared to $2\log(1/\varepsilon)$ if we require that the output be ε-close to uniform as in the Leftover Hash Lemma (Lemma 4.1). This constant loss of conditional Renyi entropy allows us to do a tighter hiding analysis in Section 6.3.1.

Lemma 6.7. *For any triple of (possibly correlated) random variables X, Y and Z,*

$$\mathrm{CP}^{1/2}(X|Y) \le \mathrm{CP}^{1/2}(X|(Y,Z)) \le \sqrt{|\mathrm{Supp}(Z)|}\cdot \mathrm{CP}^{1/2}(X|Y) \ .$$

This says that conditioning on random variable Z can only decrease the conditional Renyi entropy, but does so by at most $\log(|\mathrm{Supp}(Z)|)$ bits. For Shannon entropy, a stronger statement can be proven, namely that conditioning on Z reduces entropy by at most $\mathrm{H}(Z)$ bits: $\mathrm{H}(X|Z) \ge \mathrm{H}(X) - \mathrm{H}(Z)$ (omitting the random variable Y for simplicity). This follows from the chain rule $\mathrm{H}(X,Z) = \mathrm{H}(Z) + \mathrm{H}(X|Z)$, noting that $\mathrm{H}(X,Z) \ge \mathrm{H}(X)$. However, the chain rule does not hold for conditional Renyi entropy.[9]

The final lemma is a stronger variant of the previous Leftover Hash Lemma of Lemma 4.1, with its hypothesis stated in terms of collision probability.

Lemma 6.8. *(Leftover Hash Lemma, strengthened [37, 5].) Let random variable H denote a random hash function from a family of pairwise-independent hash functions $\mathcal{H}$ with range $\{0,1\}^\alpha$. For any $\varepsilon > 0$, if $\mathrm{CP}(X) \le \varepsilon^2 \cdot 2^{-\alpha}$ and H is independent from X, then random variable $(H, H(X))$ is ε-close in statistical distance to uniform.*

[9] In terms of collision probability, a chain rule would say $\mathrm{CP}^{1/2}(X,Z) = \mathrm{CP}^{1/2}(Z)\cdot\mathrm{CP}^{1/2}(X|Z)$. This is not true in general, as can be observed by noting that $(\mathrm{CP}^{1/2}(X,Z)/\mathrm{CP}^{1/2}(Z))^2 = \mathrm{CP}(X,Z)/\mathrm{CP}(Z)$ equals the expectation $\mathrm{CP}(X|_{Z=z})$ over z chosen according to the distribution $\hat{Z}$, where $\Pr[\hat{Z}=z] = \Pr[Z=z]^2/\mathrm{CP}(Z)$, while $\mathrm{CP}^{1/2}(X|Z)$ (and any natural definition of conditional Renyi entropy) is defined by taking an expectation over Z. By choosing a random variable Z where $\hat{Z}$ is very different from Z, and choosing appropriate conditional distributions $X|_{Z=z}$, examples can be constructed where $\mathrm{CP}^{1/2}(X,Z)$ is both much larger and much smaller than $\mathrm{CP}^{1/2}(Z)\cdot\mathrm{CP}^{1/2}(X|Z)$.

6.2.2. Average-case hiding and binding properties of interactive hashing. We now analyze the interactive hashing protocol, namely Protocol 3.3, in terms of *average-case* measures. For hiding, we use the $\mathrm{CP}^{1/2}$ measure introduced in the previous section. For the binding property, we present an average-case variant of Lemma 3.7, where we look at the *expected* number of outputs that lies in any set Γ, rather than bound the probability that there is more than one output in Γ.

LEMMA 6.9. *(Hiding of interactive hashing in* $\mathrm{CP}^{1/2}$ *measure.) Let* $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ *be the interactive hashing protocol in Protocol 3.3. If the sender* S_{IH}*'s input comes from a random variable* Y *over* $\{0,1\}^q$ *and* W *is any (possibly correlated) random variable (representing the receiver's a priori information about* Y*), then for any receiver* R^**,*

$$\mathrm{CP}^{1/2}(Z|(W,V)) \leq \sqrt{2^{q-k}} \cdot \mathrm{CP}^{1/2}(Y|W) \ ,$$

where $Z = \mathrm{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Y), R^*)(1^q, 1^k)$ *and* $V = \mathrm{view}_{R^*}(S_{\mathrm{IH}}(Y), R^*)(1^q, 1^k)$*.*

Proof. Without loss of generality, we may assume that R^* is deterministic. (The randomized case then follows by taking expectation over R^*'s coins.) Now that R^* is deterministic, the hash functions sent $h_0, \ldots, h_{q-k-1}$ by R^* are fully determined by S_{IH}'s responses $c_0, \ldots, c_{q-k-1} \in \{0,1\}$ (refer to Protocol 3.3). Hence, the number of possible different receiver's view is bounded by 2^{q-k}. This implies that $|\mathrm{Supp}(V)| \leq 2^{q-k}$, where $V = \mathrm{view}_{R^*}(S_{\mathrm{IH}}(Y), R^*)(1^q, 1^k)$. By Lemma 6.7,

$$\mathrm{CP}^{1/2}(Y|(W,V)) \leq \sqrt{|\mathrm{Supp}(V)|} \cdot \mathrm{CP}^{1/2}(Y|W) \leq \sqrt{2^{q-k}} \cdot \mathrm{CP}^{1/2}(Y|W) \ .$$

Observe that given any particular instantiation of $W = w$ and $V = v$, the distribution of $\mathrm{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Y), R^*)(1^q, 1^k)|_{W=w,V=v}$ has the same collision probability as $Y|_{W=w,V=v}$ (indeed they are in bijective correspondence). Hence, $\mathrm{CP}^{1/2}(Z|(W,V)) = \mathrm{CP}^{1/2}(Y|(W,V)) \leq \sqrt{2^{q-k}} \cdot \mathrm{CP}^{1/2}(Y|W)$. □

LEMMA 6.10. *(Binding of interactive hashing in expected measure.) Let* $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ *be the interactive hashing protocol in Protocol 3.3. For any fixed subset* $T \subseteq \{0,1\}^q$*, and for any sender* S^**, setting* $C = \mathrm{output}((S^*, R_{\mathrm{IH}})(1^q, 1^k))$*, we have*

$$\mathrm{E}\left[|\{z : C(z) \in T\}|\right] < \max\{24, 2^{k+1} \cdot \mu(T)\} \leq 24 + 2^{k+1} \cdot \mu(T) \ ,$$

where the above expectation is taken over the coins of S^* *and* R_{IH}*.*

This lemma and its proof are inspired by the work of Goldreich, Goldwasser, and Linial [38], who studied a protocol similar to interactive hashing for a different purpose (namely, random selection protocols).

Proof. Without loss of generality, we may assume that S^* is deterministic. (Else, we can fix its coins to maximize the expectation.) Note that for iteration $j = 0, \ldots, q-k-1$, R_{IH} will send a random h_j, partitioning the set of possible outputs into two sets $\{y : h_j(y) = 0\}$ and $\{y : h_j(y) = 1\}$, and S^* chooses a side of the partition by sending a bit c_j. For all $j \geq 0$, let $U_j = \{y \in \{0,1\}^q : h_i(y) = c_i \forall i < j\}$ denote the set of compatible elements of the universe at iteration j, and let $T_j = U_j \cap T$ denote the set of compatible elements of T. We write $\mu(T_j) = |T_j|/|U_j| = |T_j| \cdot 2^{-(q-j)}$ denote the density of T_j within U_j, and $\mu_j = \mathrm{E}_{[}\,[\mu]\,(T_j)]$, where the expectation is taken over random choices of $h_0, \cdots, h_{j-1}$. For convenience of notation, we denote the hash function h_i's range as $\{\pm 1\}$, instead of $\{0,1\}$.

Consider a particular set T_j, and a particular hash function h_j. Observe that for every $y \neq y' \in T_j$, $\Pr_{h_j}[h_j(y) = h_j(y')] \leq 1/2$. Hence,

$$\mathop{\mathrm{E}}_{h_j}[h_j(y)h_j(y')] \leq 0 \quad \forall y \neq y' \in T_j \ . \tag{6.1}$$

Observe that the set $T_{j+1} = \{y \in T_j : h_j(y) = c_j\}$ and thus:

$$|T_{j_1}| \leq \max_{c \in \{\pm 1\}} |\{y \in T_j : h_j(y) = c\}| = \frac{|T_j| + \left|\sum_{y \in T_j} h_j(y)\right|}{2}.$$

Therefore,

$$\begin{aligned}
\mathop{\mathrm{E}}_{h_j}[\mu(T_{j+1})] &= \mu(T_j) + 2^{-(q-j)} \cdot \mathop{\mathrm{E}}_{h_j}\left[\left|\sum_{y \in T_j} h_j(y)\right|\right] \\
&\leq \mu(T_j) + 2^{-(q-j)} \cdot \sqrt{\mathop{\mathrm{E}}_{h_j}\left[\left(\sum_{y \in T_j} h_j(y)\right)^2\right]} && \text{(Cauchy-Schwartz/Jensen)} \\
&= \mu(T_j) + 2^{-(q-j)} \cdot \sqrt{|T_j| + \sum_{y \neq y'} \mathop{\mathrm{E}}_{h_j}[h_j(y)h_j(y')]} \\
&\leq \mu(T_j) + 2^{-(q-j)} \cdot \sqrt{|T_j|} && \text{(by 6.1)} \\
&= \mu(T_j) + \sqrt{2^{-(q-j)} \cdot \mu(T_j)} \ .
\end{aligned}$$

Consequently,

$$\begin{aligned}
\mu_{j+1} &= \mathop{\mathrm{E}}_{h_0,\ldots,h_j}[\mu(T_{j+1})] \\
&= \mathop{\mathrm{E}}_{h_0,\ldots,h_{j-1}}[\mathop{\mathrm{E}}_{h_j}[\mu(T_{j+1})]] \\
&\leq \mathop{\mathrm{E}}_{h_0,\ldots,h_{j-1}}\left[\mu(T_j) + \sqrt{2^{-(q-j)} \cdot \mu(T_j)}\right] \\
&\leq \mathop{\mathrm{E}}_{h_0,\ldots,h_{j-1}}[\mu(T_j)] + \sqrt{2^{-(q-j)} \cdot \mathop{\mathrm{E}}_{h_0,\ldots,h_{j-1}}[\mu(T_j)]} && \text{(Cauchy-Schwartz/Jensen)} \\
&= \mu_j + \sqrt{2^{-(q-j)} \cdot \mu_j} \ .
\end{aligned}$$

Assume that the μ_j's are monotonically increasing (otherwise, we can make it so). This gives us

$$\begin{aligned}
\mu_{q-k} &\leq \mu_0 + \sum_{j=0}^{q-k-1} \sqrt{2^{-(q-j)} \cdot \mu_j} \\
&\leq \mu_0 + \sqrt{\mu_{q-k}} \cdot \sum_{j=0}^{q-k-1} \sqrt{2^{-(q-j)}} && \text{(μ_j's are monotonically increasing)} \\
&< \mu_0 + \sqrt{\mu_{q-k}} \cdot \sqrt{6/2^k} \\
&\leq \mu_0 + \frac{\mu_{q-k}}{2} && \text{(if $\mu_{q-k} \geq 24 \cdot 2^{-k}$)} \ ,
\end{aligned}$$

giving us $\mu_{q-k} < 2\mu_0 = 2\mu(T)$ if $\mu_{q-k} \geq 24 \cdot 2^{-k}$. This means that μ_{q-k} is either less than $24 \cdot 2^{-k}$ or less than $2\mu(T)$. Therefore, we can conclude that

$$\begin{aligned}
\mathrm{E}\left[|\{z : C(z) \in T\}| : C = \mathrm{output}((S^*, R_{\mathrm{IH}})(1^q, 1^k))\right] &= \mu_{q-k} \cdot 2^k \\
&< \max\{24, 2^{k+1} \cdot \mu(T)\} \ .
\end{aligned}$$

□

6.2.3. Protocol 5.5 is hiding in $\mathrm{CP}^{1/2}$ measure. We are now ready to analyze the hiding property of Protocol 5.5 in terms of the $\mathrm{CP}^{1/2}$ measure. To do so, we say what it means for a scheme to be δ-hiding in $\mathrm{CP}^{1/2}$ measure in Definition 6.11 below. But before going into that definition, we first establish some notations that are used throughout this part of the section.

With the sender's input being x, we let random variable $\mathrm{view}_{R^*}(S_c^1(x), R^*)$ denote the view of receiver R^* in the first commit phase, let random variable $\mathrm{output}_S(S_c^1(x), R^*)$ denote the sender's private output in the first phase, and let random variable $\mathrm{transcript}(S^1(x), R^*)$ denote the first (commit and reveal) phase transcript.

Using similar notations, with the transcript being τ and sender's input being x, we let random variable $\mathrm{view}_{R^*}(S_c^2(x), R^*)(\tau)$ denote the view of receiver R^* in the second commit phase, let random variable $\mathrm{output}_S(S_c^2(x), R^*)(\tau)$ denote the sender's private output in the second phase, and let random variable $\mathrm{transcript}(S^2(x), R^*)(\tau)$ denote the second (commit and reveal) phase transcript. We write Γ_1 in $\mathrm{view}_{R^*}(S_c^1(\Gamma_1), R^*)$—and similarly for others—to mean that the sender's private input is chosen uniformly from a set Γ_1.

Definition 6.11. *For a parameter $\delta \in [0,1]$, two-phase commitment scheme (S, R) is said to be* **δ-hiding in $\mathrm{CP}^{1/2}$ measure** *if there exists two sets $\Gamma_1, \Gamma_2 \subseteq \{0,1\}^n$ such that the following three properties hold.*

(H.1) $\Gamma_1 \cup \Gamma_2 = \{0,1\}^n$ *and* $\mu(\Gamma_1 \cap \Gamma_2) \geq \delta$.

(H.2) When the sender's private input x is chosen uniformly from Γ_1, the sender's private output in the first phase has low collision probability given the receiver's view. Formally, for any adversarial receiver R^*,

$$\mathrm{CP}^{1/2}(A|V) \leq \sqrt{2^{-(k-1)}} \ ,$$

for $(A, V) = (\mathrm{output}_S(S_c^1(\Gamma_1), R^*), \mathrm{view}_{R^*}(S_c^1(\Gamma_1), R^*))$.

(H.3) When the sender's private input x is chosen uniformly from Γ_2, the sender's private output in the second phase has low collision probability given the receiver's view. Formally, for every adversarial receiver R^* and every $\tau \in \mathrm{Supp}(\mathrm{T})$, where $\mathrm{T} = \mathrm{transcript}(S^1(\Gamma_2), R^*)$, we have

$$\mathrm{CP}^{1/2}(B_\tau|W_\tau) \leq \sqrt{2^{-(k-1)}} \ ,$$

for $(B_\tau, W_\tau) = (\mathrm{output}_S(S_c^2(\Gamma_2), R^*), \mathrm{view}_{R^*}(S_c^2(\Gamma_2), R^*))|_{\mathrm{T}=\tau}$.

Remark 6.12. *Being δ-hiding in $\mathrm{CP}^{1/2}$ measure in the above Definition 6.11 roughly means that the scheme is always hiding in at least one phase, and hiding in both phases occurs with probability δ.*

Lemma 6.13. *(Protocol 5.5 is $(1/n)$-hiding in $\mathrm{CP}^{1/2}$ measure.) Let $f\colon \{0,1\}^n \to \{0,1\}^n$ be any function (not necessarily one-way) such that $|f^{-1}(y)| \in [2n^2, 2^n/n^2]$ for every $y \in \{0,1\}^n$. There exists an integer $t_0 \in \{\lceil 2\log n \rceil, 2, \ldots, n - \lceil 2\log n \rceil\}$ such that Protocol 5.5, with setting of parameters $t = t_0$, $\Delta_1 = \Delta_2 = \lceil 2\log n \rceil$, and any $k \in \{1, \ldots, q\}$ is $(1/n)$-hiding in $\mathrm{CP}^{1/2}$ measure.*

Proof. Without loss of generality, we may assume that R^* is deterministic since we can fix the coins of R^* that maximizes the above collision probabilities. We prove that (S, R) satisfies the above three properties of Definition 6.11 as follows:

Property (H.1). Define $p(y) = \Pr[f(U_n) = y]$, and for $t \in \{\lceil 2\log n \rceil, 2, \ldots, n - \lceil 2\log n \rceil\}$, let $A_t = \{y \in \{0,1\}^n : 2^{-t} \leq p(y) < 2^{-t+1}\}$. Since $\cup_t A_t = f(\{0,1\}^n)$, there exists an index t_0 such that $\Pr[f(U_n) \in A_{t_0}] \geq 1/n$. Define sets Γ_1 and Γ_2 as follows:

$$\begin{aligned}\Gamma_1 &= \{x : p(f(x)) < 2^{-t_0+1}\} \\ \Gamma_2 &= \{x : p(f(x)) \geq 2^{-t_0}\}\end{aligned}$$

By the definition of Γ_1 and Γ_2, we have that $\mu(\Gamma_1 \cap \Gamma_2) = \Pr[f(U_n) \in A_{t_0}] \geq 1/n$, and also $\Gamma_1 \cup \Gamma_2 = \{0,1\}^n$.

Property (H.2). In the case when the sender's private input $x \in \Gamma_1$, we bound the collision probability of the first phase secret as follows:

$$\begin{aligned}\mathrm{CP}(f(\mathit{\Gamma}_1)) &= \sum_{y \in f(\Gamma_1)} \left(\frac{p(y)}{\mu(\Gamma_1)}\right)^2 \\ &\leq \left(\max_{y \in f(\Gamma_1)} p(y)\right) \cdot \left(\sum_{y \in f(\Gamma_1)} p(y)\right) \cdot \frac{1}{\mu(\Gamma_1)^2} \\ &< 2^{-t_0+1} \cdot \mu(\Gamma_1) \cdot \mu(\Gamma_1)^{-2} \\ &\leq 2^{-(t_0 - \log n - 1)} \qquad\qquad (\text{since } \mu(\Gamma_1) \geq 1/n) \ .\end{aligned}$$

Observe that $\mathrm{CP}(f(\mathit{\Gamma}_1)) \leq 2^{-(t_0 - \log n - 1)} \leq 2^{-(t_0 - \Delta_1 + 3)}$. Therefore we can apply Randomness Extraction Lemma 6.6 to get $\mathrm{CP}^{1/2}(Z^{(1)}) \leq \sqrt{2^{-(q-1)}}$, where $Z^{(1)} = (H_1, H_1(f(\mathit{\Gamma}_1)))$ and H_1 is an independent random hash from $\mathcal{H}_1$.

Next, let $A = \mathrm{output}_S(S_c^1(\mathit{\Gamma}_1), R^*)$ denote the private output of the sender S in the first phase of Protocol 5.5, which in turn is equal to the output of S_{IH} in the interactive hashing protocol, so equivalently $A = \mathrm{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Z^{(1)}), R^*)$. Similarly, let $V = \mathrm{view}_{R^*}(S_c^1(\mathit{\Gamma}_1), R^*)$ denote the view of the adversarial receiver R^* in the first phase, which in turn is equal to the view of R^* in the interactive hashing protocol, so equivalently $V = \mathrm{view}_{R^*}(S_{\mathrm{IH}}(Z^{(1)}), R^*)$.

The final step is to use the hiding property of interactive hashing given by Lemma 6.9 to bound the collision probability of A (the private output of the sender S) given V (the view of the adversarial receiver R^*) as follows:

$$\mathrm{CP}^{1/2}(A|V) \leq \sqrt{2^{q-k}} \cdot \sqrt{\mathrm{CP}(Z^{(1)})} \leq \sqrt{2^{q-k}} \cdot \sqrt{2^{-(q-1)}} = \sqrt{2^{-(k-1)}} \ .$$

Property (H.3). In the case when the sender's private input $x \in \Gamma_2$, we analyze the collision probability of the second phase secret as follows. First we observe that for any $x, x' \in \{0,1\}^n$ such that $f(x) = f(x')$, the first phase transcripts for both x and x' are identically distributed, that is $\mathrm{transcript}(S^1(x), R^*) \equiv \mathrm{transcript}(S^1(x'), R^*)$. Thus, if we fix a first phase transcript $\tau \in \mathrm{transcript}(S^1(x), R^*)$ containing a value $y = f(x)$ in the reveal phase, any element in $\Gamma_{2,y} = f^{-1}(y) \subseteq \Gamma_2$ is equally likely to have generated τ. Also observe that the $\Gamma_{2,y}$'s form a partition of Γ_2.

Note that by definition, $|\Gamma_{2,y}| \geq 2^{n-t_0}$, and hence $\mathrm{CP}(\mathit{\Gamma}_{2,y}) \leq 2^{-(n-t_0)} \leq 2^{-(n-t_0-\Delta_2+3)}$. Therefore we can apply Randomness Extraction Lemma 6.6 to get $\mathrm{CP}^{1/2}(Z^{(2)}) \leq \sqrt{2^{-(q-1)}}$, for $Z^{(2)} = (H_2, H_2(\mathit{\Gamma}_{2,y}))$.

Next, let $B_\tau = \mathrm{output}_S(S_c^2(\mathit{\Gamma}_{2,y}), R^*)(\tau)$ denote the private output of the sender S in the second phase, which in turn is equal to the output of S_{IH} in the interactive hashing protocol, so equivalently $B_\tau = \mathrm{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Z^{(2)}), R^*)$. Similarly, let $W_\tau = \mathrm{view}_{R^*}(S_c^2(\Gamma_{2,y}), R^*)(\tau)$ denote the view of the adversarial receiver R^* in the second phase, which in turn is equal to the view of R^* in the interactive hashing protocol, so equivalently $W_\tau = \mathrm{view}_{R^*}(S_{\mathrm{IH}}(Z^{(2)}), R^*)$.

The final step is to use the hiding property of interactive hashing given by Lemma 6.9 to bound the collision probability of B_τ (the private output of the sender S) given W_τ (the view of the adversarial receiver R^*) as follows:

$$\mathrm{CP}^{1/2}(B_\tau | W_\tau) \leq \sqrt{2^{q-k}} \cdot \sqrt{\mathrm{CP}(Z^{(2)})} \leq \sqrt{2^{q-k}} \cdot \sqrt{2^{-(q-1)}} = \sqrt{2^{-(k-1)}} \ .$$

□

6.2.4. Protocol 5.5 is 1-out-of-2 binding in expected measure. The definition of 1-out-of-2 binding in Definition 5.4 considers the first phase (resp., second phase) to be broken if the sender S^* produces valid decommitments to *two* different values after the first commit stage (resp., second commit stage). In this section and Section 6.3, we will work with a relaxed notion where we simply bound the *expected* number of values to which a cheating sender can open. To this end, we define openings(S^*, R^1) [resp., openings(S^*, R^2)] to be a random variable denoting the set of values to which the sender successfully opens in phase 1 [resp., phase 2], where 'successfully' opens is defined for each phase analogously to Definition 5.4. (Definition 5.4 refers to sender strategies that produce decommitments to one or two values in each reveal phase; here we consider a natural generalization where the sender tries to decommit to many values in each phase.) More formally, for a two-phase commitment scheme (S, R) and a 'binding' set $\mathcal{B}$, we define openings$(S^*, R^1)(\mathcal{B})$ as follows:

- S^* and R_c^1 interact to get first phase commitment $c^{(1)}$.
- After the interaction, S^* outputs a sequence of values $d_1^{(1)}, \ldots, d_\ell^{(1)}$ and corresponding full transcripts $\lambda_1, \ldots, \lambda_\ell$ of *both* phases. Recall that $\lambda_i = (\tau_i, \kappa_i)$, where τ_i and κ_i are the first-phase and second-phase transcripts, respectively.
- We let openings$(S^*, R^1)(\mathcal{B})$ be the set of distinct values $d_i^{(1)}$ whose opening λ_i is valid, where by valid we mean that λ_i begins with prefix $c^{(1)}$, λ_i contains a decommitment of $c^{(1)}$ to the value $d_i^{(1)}$ with first-phase transcript $\tau_i \notin \mathcal{B}$, and both R_r^1 and R_r^2 accept in λ_i.

Analogously, we define openings$(S^*, R^2)(\tau)$, where τ is a first-phase transcript, as follows:

- S^* and R_c^2 interact to get second phase commitment $c^{(2)}$.
- After the interaction, S^* outputs a sequence of values $d_1^{(2)}, \ldots, d_\ell^{(2)}$ and corresponding full transcripts $\lambda_1, \ldots, \lambda_\ell$.
- We let openings$(S^*, R^2)(\tau)$ be the set of distinct values $d_i^{(2)}$ whose opening λ_i is valid, where by valid we mean that λ_i starts with prefix $c^{(2)}$ (which in turn contains τ as a prefix), λ_i contains a decommitment of $c^{(2)}$ to the value $d_i^{(2)}$, and R_r^2 accepts in λ_i.

Now, we can describe the binding property of Protocol 5.5 in this language (even when the underlying one-way function has unknown security).

LEMMA 6.14. *(Protocol 5.5 is 1-out-of-2 binding in expected measure.) For every integer $t \in \{1, 2, \ldots, n\}$, $k = O(\log n)$, $\Delta_1 = O(\log n)$, and $\Delta_2 = O(\log n)$, the following holds for the two-phase commitment scheme (S, R) in Protocol 5.5 based on one-way function $f\colon \{0,1\}^n \to \{0,1\}^n$:*

There exists a binding set $\mathcal{B}$ for (S, R) where:

(B.1) No PPT adversary S^* can break the first phase binding with nonnegligible probability in the sense of Definition 5.4. That is, for every PPT S^*, we have $|\operatorname{openings}(S^*, R^1)(\mathcal{B})| \leq 1$ with probability $1 - \operatorname{neg}(n)$ over the coins of S^* and R^1_c.

(B.2) For all $\tau \in \mathcal{B}$ and every adversarial sender S^*,

$$\mathrm{E}\left[\left|\operatorname{openings}(S^*, R^2)(\tau)\right|\right] < 2 \ ,$$

where the above expectation is taken over the coins of S^* and R^2.

Proof. We follow the proof of the binding property in Lemma 5.9, using both Claims 5.11 and 5.10 from that proof. Let $\mathcal{B} = \{y \in \{0,1\}^n : \Pr[f(U_n) = y] \leq 2^{-t-\Delta_3}\}$ be the same binding set as defined in both claims. We set $\Delta_3 = \Delta_2 + O(\log n)$ to be large enough so that the binding parameter $\operatorname{poly}(n) \cdot 2^{-\Omega(\Delta_3 - \Delta_2)} \cdot 2^{k/2}$ in Claim 5.11 is at most 2^{-k}. (This can be done since $k = O(\log n)$.) Now, Claim 5.11 states that if $\tau \in \mathcal{B}$, then the second commitment phase is binding. This implies that $\left|\operatorname{openings}(S^*, R^2)(\tau)\right| \geq 2$ with probability at most 2^{-k}. Since $\left|\operatorname{openings}(S^*, R^2)(\tau)\right| \leq 2^k$ (the commitment is to a k-bit string), taking expectations we have

$$\mathrm{E}\left[\left|\operatorname{openings}(S^*, R^2)(\tau)\right|\right] \leq 2^k \cdot 2^{-k} + 1 \cdot (1 - 2^{-k}) < 2 \ .$$

To see why property (B.1) holds, let $\varepsilon = \varepsilon(n)$ be the probability for which PPT S^* breaks the first phase binding. Observe that the inversion success probability of f from Claim 5.10 is

$$\begin{aligned}\varepsilon^{O(1)} \cdot 1/\operatorname{poly}(n) \cdot 2^{-(k+\Delta_1+\Delta_3)} &= \varepsilon^{O(1)} \cdot 1/\operatorname{poly}(n) \cdot 2^{-(k+\Delta_1+\Delta_2+O(\log n))} \\ &= \frac{\varepsilon^{O(1)}}{\operatorname{poly}(n)} \ ,\end{aligned}$$

since all $k, \Delta_1, \Delta_2 = O(\log n)$. This forces $\varepsilon(n)$ to be a negligible function. □

6.3. Converting weakly hiding to strongly hiding commitments. In the previous section, we established that Protocol 5.5, with appropriate choice of parameters, is $1/n$-hiding in $\mathrm{CP}^{1/2}$ measure (hence, only *weakly hiding*), and 1-out-of-2 binding in expected measure. Our goal in this section is to show how to boost the hiding probability to $\delta = 1 - \operatorname{neg}(n)$, therefore making the scheme *strongly hiding*, while maintaining the 1-out-of-2 binding property.

We first show how to double the hiding probability by combining a constant number of schemes to obtain a new scheme. We then repeat this doubling amplification process $O(\log n)$ times to boost the hiding probability from $1/n$ to a constant $c > 0$, hence obtaining an $\Omega(1)$-hiding scheme. Finally we boost it all the way to $1 - \operatorname{neg}(n)$ by combining polynomially many $\Omega(1)$-hiding schemes. This is all achieved via a hiding amplification procedure stated next. (See Section 6.1 for an overview.)

Algorithm 6.15. *Hiding amplification procedure, denoted as* Amplify.

Input: *two-phase commitment (S, R)*

Additional Input Parameters: *These are given in unary, and listed below:*

1. *Security parameter n.*
2. *Number m of schemes (S, R) to be combined.*
3. *Integer r denoting S's private input length.*
4. *Integer k denoting S's private output length.*

5. *Integer k' denoting $\mathbf{S}$'s private output length.*
6. *Integer thresholds α_1 and α_2, for the first and second commit phases respectively.*

Output: *two-phase commitment $(\mathbf{S}, \mathbf{R})$, as described by Protocol 6.16.*

To reduce unnecessary clutter, we write $(\mathbf{S}, \mathbf{R}) = \mathsf{Amplify}(S, R)$ when the rest of the parameters are clear from context.

Protocol 6.16. *Amplified scheme $(\mathbf{S}, \mathbf{R})$ from hiding amplification of base scheme (S, R).*

Sender's private input: $x = (x_1, \ldots, x_m) \in \{0,1\}^{mr}$.

First phase commit:

1. *$(\mathbf{S}_c^1, \mathbf{R}_c^1)$ do m sequential executions of (S_c^1, R_c^1), using x_i for S_c^1's private input in the i-th execution. Let $(S_c^1[i](x_i), R_c^1[i])$ denote the i-th execution of (S_c^1, R_c^1). Define $a_i = \text{output}_S(S_c^1[i](x_i), R_c^1[i]) \in \{0,1\}^k$, and let $a = (a_1, \ldots, a_m)$.*
2. *Let $\mathcal{H}_1 = \{h_1 \colon \{0,1\}^{mk} \to \{0,1\}^{\alpha_1}\}$ be a family of pairwise independent hash functions. $\mathbf{S}^1$ chooses a random hash $h_1 \leftarrow \mathcal{H}_1$, and computes $y^{(1)} = (h_1, h_1(a)) \in \{0,1\}^q$.*
3. *$(\mathbf{S}_c^1, \mathbf{R}_c^1)$ run the interactive hashing protocol $(S_{\text{IH}}^1(y^{(1)}), R_{\text{IH}}^1)(1^q, 1^{k'})$, given by Protocol 3.3, with $\mathbf{S}^1$ and $\mathbf{R}^1$ acting as S_{IH}^1 and R_{IH}^1, respectively.*
   Let circuit $C \colon \{0,1\}^{k'} \to \{0,1\}^q$ be the common output, and $d^{(1)} \in \{0,1\}^{k'}$ be S_{IH}^1's private output in $(S_{\text{IH}}^1(y^{(1)}), R_{\text{IH}}^1)(1^q, 1^{k'})$.

First phase sender's private output: String $d^{(1)} \in \{0,1\}^{k'}$.

First phase reveal:

S_r^1 sends tuple $\gamma^{(1)} = (d^{(1)}, a, h_1) \circ (\gamma_1^{(1)}, \ldots, \gamma_m^{(1)})$, where $\gamma_i^{(1)}$ is the first phase revelation string of $S_r^1[i]$ in the above execution of $(S_r^1[i](x_i), R_r^1[i])$.

Receiver $\mathbf{R}_r^1$ accepts if and only if $C(d^{(1)}) = (h_1, h_1(a))$ and $R_r^1[i]$ accepts $(\gamma_i^{(1)}, a_i)$ for all $i \in \{1, 2, \ldots, m\}$.

Second phase commit:

Second phase common input: Transcript τ containing $(\tau_1, \ldots, \tau_m)$, where each $\tau_i = \text{transcript}(S^1[i](x_i), R^1[i])$.

1. *$(\mathbf{S}_c^2, \mathbf{R}_c^2)$ do m sequential executions of (S_c^2, R_c^2), using x_i for S^2's secret and transcript τ_i in the i-th execution. Let $(S_c^2[i](x_i), R_c^2[i])(\tau_i)$ denote the i-th execution of (S^2, R^2). Define $b_i = \text{output}_S(S_c^2[i](x_i), R_c^2[i])(\tau_i) \in \{0,1\}^k$, and let $b = (b_1, \ldots, b_m)$.*
2. *Let $\mathcal{H}_2 = \{h_2 \colon \{0,1\}^{mk} \to \{0,1\}^{\alpha_2}\}$ be a family of pairwise independent hash functions. $\mathbf{S}^2$ chooses a random hash $h_2 \leftarrow \mathcal{H}_2$, and computes $y^{(2)} = (h_2, h_2(b)) \in \{0,1\}^q$.*
3. *$(\mathbf{S}_c^2, \mathbf{R}_c^2)$ run the interactive hashing protocol $(S_{\text{IH}}^2(y^{(2)}), R_{\text{IH}}^2)(1^q, 1^{k'})$, given by Protocol 3.3, with $\mathbf{S}_c^2$ and $\mathbf{R}_c^2$ acting as S_{IH}^2 and R_{IH}^2, respectively.*
   Let circuit $C \colon \{0,1\}^{k'} \to \{0,1\}^q$ be the common output, and $d^{(2)} \in \{0,1\}^{k'}$ be S_{IH}^2's private output in $(S_{\text{IH}}^2(y^{(2)}), R_{\text{IH}}^2)(1^q, 1^{k'})$.

Second phase sender's private output: String $d^{(2)} \in \{0,1\}^{k'}$.

Second phase reveal:

S_r^2 sends tuple $\gamma^{(2)} = (d^{(2)}, b, h_2) \circ (\gamma_1^{(2)}, \ldots, \gamma_m^{(2)})$, where $\gamma_i^{(2)}$ is the second phase revelation string of $S_r^2[i]$ in the above execution of $(S_r^2[i](x_i), R_r^2[i])$.

Receiver $\mathbf{R}_r^2$ accepts if and only if $C^{(2)}(d^{(2)}) = (h_2, h_2(b))$ and $R_r^2[i]$ accepts

$(\gamma_i^{(2)}, b_i)$ for all $i \in \{1, 2, \ldots, m\}$.

Starting from a weakly hiding scheme (S_0, R_0) of Protocol 5.5, we iteratively apply the amplification process $\mathsf{Amplify}$, in a way described by Algorithm 6.17 below, to achieve a new scheme (S, R) that we will show to be statistically hiding. Let $D > 1$ denote a large enough integer constant. We will set the number of schemes to be combined to be $m = D$ in all but the last iteration, in which we set $m = n$.

ALGORITHM 6.17. *Iterative amplification procedure.*

Input: *Function $f : \{0,1\}^n \to \{0,1\}^n$, constant integer $D > 1$, and thresholds $t \in \{1, 2, \ldots, n\}$, $\beta_1, \ldots, \beta_\ell \in \{0, 1, \ldots, D-1\}$, $\beta_{\ell+1} \in \{0, 1, \ldots, n\}$.*

1. *Let $\ell = \lceil \log n \rceil$, $k_0 = (16D) \cdot \ell$, and (S_0, R_0) be the two-phase commitment scheme based on function $f\colon \{0,1\}^n \to \{0,1\}^n$ from Protocol 5.5 using parameters t, $k = k_0$, and $\Delta_1 = \Delta_2 = \lceil 2 \log n \rceil$.*
2. *For $j = 1, 2, \ldots, \ell$, repeat the following:*
    - *(a) Set $k_j = k_{j-1} - 8D - 8$.*
    - *(b) Set $(S_j, R_j) = \mathsf{Amplify}(S_{j-1}, R_{j-1})$ for settings of parameters $m = D$, $r = n \cdot D^{j-1}$, $k = k_{j-1}$, $k' = k_j$, $\alpha_1 = (\beta_j + 1)(k_{j-1} - 1) - 3$ and $\alpha_2 = (D - \beta_j)(k_{j-1} - 1) - 3$.*
3. *Set $(\mathsf{S}, \mathsf{R}) = \mathsf{Amplify}(S_\ell, R_\ell)$ for settings of parameters $m = n$, $r = n \cdot D^\ell$, $k = k_\ell$, $k' = n$, $\alpha_1 = \lfloor (\beta_{\ell+1} + \frac{1}{3}\delta n)k \rfloor$ and $\alpha_2 = \lfloor (n - \beta_{\ell+1} + \frac{1}{3}\delta n)k \rfloor$, where $\delta = 1/D$.*

Output: *two-phase commitment scheme (S, R).*

LEMMA 6.18. *If scheme (S_0, R_0) used by Algorithm 6.17 runs in polynomial time, then scheme (S, R), the output of Algorithm 6.17, also runs in polynomial time.*

Proof. Scheme (S, R) consists of $n \cdot D^\ell = n \cdot D^{O(\log n)} = \mathrm{poly}(n)$ executions of (S_0, R_0). In addition, each amplification procedure $\mathsf{Amplify}$ adds an overhead time of $\mathrm{poly}(n)$ since both the sender and receiver are doing interactive hashing. Since there are only $1 + n + nD + nD^2 + \cdots + D^{\ell-1} = \mathrm{poly}(n)$ amplifications steps, the overhead time is polynomial. Hence, scheme (S, R) runs in polynomial time if (S_0, R_0) does too. □

The rest of this section, which is technically involved, is devoted to proving the hiding and binding properties of the final scheme (S, R). (In process of doing so, we also analyze the hiding and binding properties of intermediate schemes (S_j, R_j).)

6.3.1. Hiding amplification. The following two lemmas, Lemma 6.19 and 6.20, provide us a way to understand the hiding property (in the $\mathrm{CP}^{1/2}$ measure) of amplified scheme $(\mathbf{S}, \mathbf{R})$, in terms of its base scheme (S, R). Lemma 6.19 basically say that the hiding probability doubles when we go from (S_{j-1}, R_{j-1}) to $(S_j, R_j) = \mathsf{Amplify}(S_{j-1}, R_{j-1})$ (refer to Step 2b in Algorithm 6.17). So if we start up with $1/n$-hiding scheme (S_0, R_0), in $\ell = \log n$ iterations, we will get a scheme (S_ℓ, R_ℓ) with $\Omega(1)$-hiding. Lemma 6.20 essentially argues that the final amplification step boost the hiding probability all the way to $1 - \mathrm{neg}(n)$ (in both phases) when starting from a scheme that is $\Omega(1)$-hiding. With these two lemmas, we can establish that the final scheme $(\mathsf{S}, \mathsf{R}) = \mathsf{Amplify}(S_\ell, R_\ell)$ is statistically hiding in both phases.

LEMMA 6.19. *(Intermediate step hiding amplification.) For every sufficiently large constant $D \in \mathbb{Z}^+$, the following holds: If scheme (S, R) is δ-hiding in $\mathrm{CP}^{1/2}$ measure, then there exists an integer $\beta \in \{0, 1, \ldots, D-1\}$ such that scheme $(\mathbf{S}, \mathbf{R}) = \mathsf{Amplify}(S, R)$, with parameters $m = D$, $k' = k - 8D - 8$, $\alpha_1 = (\beta + 1)(k - 1) - 3$, and $\alpha_2 = (D - \beta)(k - 1) - 3$, is δ'-hiding in $\mathrm{CP}^{1/2}$ measure, for $\delta' = \min\{2\delta, 1/D\}$.*

Proof. Without loss of generality, we may assume that R^* is deterministic since we can fix the coins of R^* that maximizes the collision probability. Throughout this proof, the value of m will be fixed to D, although we will keep writing m. Let the δ-hiding properties, as stated in Definition 6.11, of (S, R) be (H.1), (H.2) and (H.3), respectively. We will prove that $(\mathbf{S}, \mathbf{R})$ satisfies Definition 6.11 with Properties (H'.1), (H'.2) and (H'.3) by showing that Property (H.1) implies (H'.1), and so forth.

Property (H.1) implies (H'.1). Let Γ_1 and Γ_2 be the corresponding sets for (S, R). Define the sets Γ_1' and Γ_2' in terms as follows (the value of β will be determined later).

$$\begin{aligned} \Gamma_1' &= \{(x_1, \ldots, x_m) : \exists\, i_1, \ldots, i_{\beta+1} \text{ such that } x_{i_1}, \ldots, x_{i_{\beta+1}} \in \Gamma_1\} \ , \\ \Gamma_2' &= \{(x_1, \ldots, x_m) : \exists\, i_1, \ldots, i_{m-\beta} \text{ such that } x_{i_1}, \ldots, x_{i_{m-\beta}} \in \Gamma_2\} \ . \end{aligned}$$

By the way we defined Γ_1' and Γ_2' together with the fact that $\Gamma_1 \cup \Gamma_2 = \{0,1\}^r$, it is the case that $\Gamma_1' \cup \Gamma_2' = \{0,1\}^{mr}$. This is because either at least $\beta + 1$ of the x_i are in Γ_1 (in which case, $(x_1, \ldots, x_m) \in \Gamma_1'$) or else at most β of the x_i are in Γ_1, which implies that at least $m - \beta$ of the x_i are in Γ_2 (in which case, $(x_1, \ldots, x_m) \in \Gamma_2'$).

We are given that $\mu(\Gamma_1 \cap \Gamma_2) \geq \delta$. Define $\delta' = \min\{\delta, 1/(2m)\}$. What we need to show is that $\mu(\Gamma_1' \cap \Gamma_2') \geq \delta'$. Choose any subset $S \subseteq \Gamma_1 \cap \Gamma_2$ such that $\mu(S) = \delta'$. Hence, we have

$$\Pr_{x_1, \ldots, x_m \leftarrow \{0,1\}^r}[\text{exactly one } x_i \in S] = m\delta'(1-\delta')^{m-1} \geq m\delta'(1-1/(m-1))^{m-1} = \Omega(m\delta') \ .$$

Given that exactly one $x_i \in S$, assume without loss of generality that $x_m \in S$. Let p_t denote the conditional probability that exactly t of the rest of the $m - 1$ x_i's are in $\Gamma_1 \setminus \Gamma_2$. Choose $\beta \in [0, m-1]$ to maximize p_t, i.e., $\beta = \operatorname{argmax}_t p_t$. Let I_i, for $i = 1, 2, \ldots, m-1$, be a binary random variable indicating whether $x_i \in \Gamma_1$ or not; note that these are independent random variables conditioned on the fact that $x_m \in S$. Let the μ the mean of the I_i's. By a Chernoff bound,

$$\Pr\left[\left|\sum_i I_i - \mu \cdot (m-1)\right| > 3\sqrt{m-1}\right] \leq 2e^{((m-1)/3)\cdot(3/\sqrt{m-1})^2} < 1/2 \ .$$

This means that greater 1/2 of the probability mass is in the interval $\mu \cdot (m-1) \pm 3\sqrt{m-1}$. Since we chose $\beta = \operatorname{argmax}_t p_t$ in a maximal way, we have

$$\Pr_{x_1, \ldots, x_m \leftarrow \{0,1\}^r}[\text{exactly } \beta \text{ of } x_i\text{'s are in } \Gamma_1 \setminus S \mid \text{exactly one } x_i \in S] = \Omega\left(\frac{1}{\sqrt{m}}\right) \ .$$

Knowing that $\Gamma_1 \cup \Gamma_2 = \{0,1\}^r$, if exactly β of x_i's in $\Gamma_1 \setminus S$ and exactly one $x_i \in S$, then there must be at least $m - 1 - \beta$ of x_i's in $\Gamma_2 \setminus S$. Consequently,

$$\begin{aligned} \Pr_{x_1, \ldots, x_m \leftarrow \{0,1\}^r}[(x_1, \ldots, x_m) \in \Gamma_1' \cap \Gamma_2'] &= \Omega(m\delta') \cdot \Omega\left(\frac{1}{\sqrt{m}}\right) \\ &= \Omega(\sqrt{m}\delta') \\ &> 2\delta' = \min\{2\delta, 1/m\}, \end{aligned}$$

where the last inequality holds when $m = D$ is a large enough constant.

Property (H.2) implies (H'.2). In the first commitment phase $(\mathbf{S}_c^1, R^*)$, the cheating receiver R^* interacts with m sequential executions of S_c^1. Here we must analyze the case when S_c^1's private input in these m executions, given by $x = (x_1, \ldots, x_m)$, is distributed uniformly in Γ_1'. We let $A_i(x)$ denote the private output of the sender and $V_i(x)$ the view of the receiver in the i'th execution, for x being the private input for $\mathbf{S}_c^1$. That is, for $i = 1, \ldots, m$,

$$\begin{aligned} A_i(x) &= \mathrm{output}_S(S_c^1(x_i), R^*(V_1, \ldots, V_{i-1})); \\ V_i(x) &= \mathrm{view}_{R^*}(S_c^1(x_i), R^*(V_1, \ldots, V_{i-1})). \end{aligned}$$

Note that while the sender's behavior in the i'th execution is independent of the previous executions, the cheating receiver may base its strategy on its previous views. We want to bound $\mathrm{CP}^{1/2}(A''(\varGamma_1')|V''(\varGamma_1'))$, where $A''(\varGamma_1') = (A_1(\varGamma_1'), \ldots, A_m(\varGamma_1'))$ represents the combined first-phase private outputs of the m senders, and $V''(\varGamma_1') = (V_1(\varGamma_1'), \ldots, V_m(\varGamma_1'))$ represents the view of R^* when interacting with these m senders. Note that random variable $\varGamma_1'$ represents an independent random element from the set Γ_1'. To do this, we consider, for each $I \subseteq [m]$ of size at least $\beta + 1$, the random variable $\varGamma_1'|_I$ for private input of the sender $\mathbf{S}$, where $\varGamma_1'|_I$ represents choosing x_i uniformly in Γ_1 for $i \in I$, and uniformly in $\overline{\Gamma_1}$ for $i \notin I$. To get a bound on $\mathrm{CP}^{1/2}(A''(\varGamma_1'|_I)|V''(\varGamma_1'|_I))$, we will have to refer to Lemma 6.5 and see why the (A_i, V_i)'s satisfy the two conditions of the lemma.

Conditioned on the any previous view—namely, $V_1(\varGamma_1'|_I) = v_1, \ldots, V_{i-1}(\varGamma_1'|_I) = v_{i-1}$ for any $v_1, \ldots, v_{i-1}$—it is the case that $\mathrm{CP}^{1/2}(A_i(\varGamma_1'|_I)|V_i(\varGamma_1'|_I)) \leq \sqrt{2^{-(k-1)}}$ if $i \in I$. This follows from Property (H.2) because the receiver R^* can incorporate the previous view $v_1, \ldots, v_{i-1}$ as advice (since R^* is unbounded), and then the only randomness in the definition of A_i and V_i is the sender's coins $x_i \leftarrow (\varGamma_1'|_I)_i$, which are uniform in Γ_1 (even conditioned on $v_1, \ldots, v_{i-1}$). This shows that the first condition of Lemma 6.5 is satisfied.

For the second condition, what we need to show is that conditioned on $V_1(\varGamma_1'|_I) = v_1, \ldots, V_i(\varGamma_1'|_I) = v_i$, the random variables $A_1(\varGamma_1'|_I), \ldots, A_i(\varGamma_1'|_I), V_{i+1}(\varGamma_1'|_I)$ are independent. This can be seen by induction on i as follows. It is vacuously true for $i = 0$. Assuming it is true for $i = j - 1$, we prove it for $i = j$ as follows. First condition on $v_1, \ldots, v_{j-1}$. By inductive hypothesis, $A_1, \ldots, A_{j-1}, V_j$ are independent (omitting $\varGamma_1'|_I$ from the notation for readability). Moreover, since we have conditioned on $v_1, \ldots, v_{j-1}$, we have that A_j and V_j are functions of only $(\varGamma_1'|_I)_j$, the sender's coins in the j'th execution, which is independent of $A_1, \ldots, A_{j-1}$ (because we have only used $(\varGamma_1'|_I)_1, \ldots, (\varGamma_1'|_I)_{j-1}$ so far). Thus, if we condition on $V_j = v_j$, A_j remains independent of $A_1, \ldots, A_{j-1}$. V_{j+1} is independent of $A_1, \ldots, A_j$ because now it is only a function of $(\varGamma_1'|_I)_{j+1}$, which has not been used yet.

Applying Lemma 6.5, we have

$$\mathrm{CP}^{1/2}(A''(\varGamma_1'|_I)|V''(\varGamma_1'|_I)) \leq \sqrt{2^{-(\beta+1)(k-1)}}, \tag{6.2}$$

since from property (H.2), it is the case that for all $i \in I$, $\mathrm{CP}^{1/2}(A_i|V_i) \leq \sqrt{2^{-(k-1)}}$ (even conditioned on the previous views), and $|I| \geq \beta + 1$.

Now, to bound $\mathrm{CP}^{1/2}(A''(\varGamma_1')|V''(\varGamma_1'))$ where X is uniform in Γ_1', we observe that $\varGamma_1' = \varGamma_1'|_{\mathcal{I}}$, where $\mathcal{I}$ is the random variable on subsets I of size at least $\beta + 1$ given by

$$\Pr[\mathcal{I} = I] = \Pr_{(x_1, \ldots, x_m) \leftarrow \Gamma_1'}[\{i : x_i \in \Gamma_1\} = I].$$

In other words, sampling from Γ'_1 can be broken into two steps; first sampling an $I \leftarrow \mathcal{I}$, and then sampling $x_i \leftarrow \Gamma_1$ for $i \in I$, and $x_i \leftarrow \overline{\Gamma_1}$ for $i \notin I$. Therefore, we have

$$\begin{aligned}
\mathrm{CP}^{1/2}(A''(\Gamma'_1|_{\mathcal{I}})|V''(\Gamma'_1|_{\mathcal{I}})) &\leq \mathrm{CP}^{1/2}(A''(\Gamma'_1|_{\mathcal{I}})|(V''(\Gamma'_1|_{\mathcal{I}}),\mathcal{I})) \qquad \text{(by Lemma 6.7)}\\
&= \mathop{\mathrm{E}}_{I \leftarrow \mathcal{I}}\left[\mathrm{CP}^{1/2}(A''(\Gamma'_1|_I)|V''(\Gamma'_1|_I)\right]\\
&\leq \sqrt{2^{-(\beta+1)(k-1)}} \qquad\qquad (6.3)\\
&= \sqrt{2^{-(\alpha_1+3)}},
\end{aligned}$$

with the last inequality following from (6.2). Therefore we can apply Randomness Extraction Lemma 6.6 to get $\mathrm{CP}^{1/2}(H_1, H_1(A''(\Gamma'_1))|V''(\Gamma'_1)) \leq \sqrt{2^{-(q-1)}}$, where H_1 is an independent random hash from $\mathcal{H}_1$.

Next, let $A' = \mathrm{output}_{\mathbf{S}}(\mathbf{S}^1(\Gamma'_1), R^*)$ denote the private output of the sender $\mathbf{S}$ in the first phase, which in turn is equal to the output of S_{IH} in the interactive hashing protocol, so equivalently $A' = \mathrm{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q), R^*_{\mathrm{IH}})$, where $Q = (H_1, H_1(A''(\Gamma'_1)))$. Similarly, let $V' = \mathrm{view}_{R^*}(\mathbf{S}^1(\Gamma'_1), R^*)$ denote the view of the adversarial receiver R^* in the first phase, which in turn is equal to the view of R^* in the interactive hashing protocol together with V'', so equivalently $V' = (\mathrm{view}_{R^*_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q), R^*_{\mathrm{IH}}), V'')$, for the same $Q = (H_1, H_1(A''(\Gamma'_1)))$.

The final step is to use the hiding property of interactive hashing given by Lemma 6.9 to bound the collision probability of A' (the private output of the sender $\mathbf{S}$) given V' (the view of the adversarial receiver R^*) as follows:

$$\mathrm{CP}^{1/2}(A'|V') \leq \sqrt{2^{q-k'}} \cdot \mathrm{CP}^{1/2}(Q|V'') \leq \sqrt{2^{q-k'}} \cdot \sqrt{2^{-(q-1)}} = \sqrt{2^{-(k'-1)}}\ .$$

Property (H.3) implies (H'.3). Fix a transcript $\tau' \in \mathrm{Supp}(\mathrm{T}')$, where random variable $\mathrm{T}' = \mathrm{transcript}(\mathbf{S}^1(\Gamma'_2), R^*)$. Transcript τ' contains first-phase transcripts $(\tau_1, \ldots, \tau_m)$ for the m executions of (S, R). Similarly to the above proof of Property (H'.2), we define the following random variables:

$$\begin{aligned}
B_i(x) &= \mathrm{output}_S(S^2_c(x_i), R^*(W_1, \ldots, W_{i-1})(\tau_i));\\
W_i(x) &= \mathrm{view}_{R^*}(S^2_c(x_i), R^*(W_1, \ldots, W_{i-1})(\tau_i)),
\end{aligned}$$

where x_i is the private input of the sender in the i'th execution of the (S, R). As above, we want to bound $\mathrm{CP}^{1/2}(B''(X_{\tau'})|W''(X_{\tau'}))$, where random variable $B''(X_{\tau'}) = (B_1(X_{\tau'}), \ldots, B_m(X_{\tau'}))$ represents the combined second-phase private outputs of the m senders, and random variable $W''(X_{\tau'}) = (W_1(X_{\tau'}), \ldots, W_m(X_{\tau'}))$ represents the view of R^* when interacting with these m senders, with $X_{\tau'}$ being a shorthand for $\Gamma'_2|_{\mathrm{T}(\Gamma'_2)=\tau'}$. To do this, we consider, for each subset $J \subseteq [m]$ of size at least $m-\beta$, the random variable X_J for private input of the sender $\mathbf{S}$, where X_J represents choosing x_i uniformly in Γ_2 for $i \in J$, and uniformly in $\overline{\Gamma_2}$ for $i \notin J$.

It is important to note that even when we condition on $\mathrm{T}'(X_J) = \tau'$, the components $(X_1, \ldots, X_m)$ of X_J remain independent, and the distribution of $X_i|_{\mathrm{T}'(X_J)=\tau'}$ is equivalent to $X_i|_{\mathrm{T}(X_i)=\tau_i}$, where only condition on the transcript of the i'th execution. (Similarly to the inductive proof above, it can be shown that $(X_1, \ldots, X_m)$ are independent given the receiver's view V_m of the m executions of S^1_c. The only additional information revealed about the X_i's in the first phase is $(A_1, \ldots, A_m)$, where A_i is a function only of X_i once we condition on V_m.)

Thus from property (H.3), we have for all $i \in J$, $\mathrm{CP}^{1/2}(B_i(X_{J,\tau'})|W_i(X_{J,\tau'})) \leq \sqrt{2^{-(k-1)}}$, where $X_{J,\tau'} = \Gamma_2'|_J|_{\mathrm{T}'(\Gamma_2'|_J)=\tau'}$, and this holds even conditioned on the previous views. Similar to the first phase, we apply Lemma 6.5 to show that

$$\mathrm{CP}^{1/2}(B''(X_{J,\tau'})|W''(X_{J,\tau'})) \leq \sqrt{2^{-(m-\beta)(k-1)}} \ .$$

Again analogous to the first phase, we observe that $X_{\tau'} = X_{\mathcal{J},\tau'}$ for an appropriate random variable $\mathcal{J}$ on sets of size at least $m-\beta$, and thus

$$\begin{aligned} \mathrm{CP}^{1/2}(B''(X_{\tau'})|W''(X_{\tau'})) &\leq \sqrt{2^{-(m-\beta)(k-1)}} \\ &= \sqrt{2^{-(\alpha_2+3)}}. \end{aligned} \tag{6.4}$$

By the Randomness Extraction Lemma 6.6, we get $\mathrm{CP}^{1/2}(H_2, H_2(B''(X_{\tau'}))|W''(X_{\tau'})) \leq \sqrt{2^{-(q-1)}}$.

The final step is to use the hiding property of interactive hashing given by Lemma 6.9 to bound the collision probability of B_τ (the private output of the sender S) given W_τ (the view of the adversarial receiver R^*) as follows:

$$\mathrm{CP}^{1/2}(B'_{\tau'}|W'_{\tau'}) \leq \sqrt{2^{q-k'}} \cdot \sqrt{2^{-(q-1)}} = \sqrt{2^{-(k'-1)}} \ .$$

□

Lemma 6.20. *(Final step hiding amplification.) The following statement holds for every constant $\delta > 0$ and every integer $k \geq 100/\delta$: If scheme (S, R) is δ-hiding in* $\mathrm{CP}^{1/2}$ *measure, then there exists an integer $\beta \in [0, n]$ such that scheme* $(\mathbf{S}, \mathbf{R}) = \mathsf{Amplify}(S, R)$, *with parameters $m = n$, $k' = n$, $\alpha_1 = \lfloor(\beta + \frac{1}{3}\delta n)k\rfloor$ and $\alpha_2 = \lfloor(n - \beta + \frac{1}{3}\delta n)k\rfloor$, is statistically hiding in the sense of Definition 5.3.*

Proof. Let the δ-hiding properties, as stated in Definition 6.11, of (S, R) be (H.1), (H.2) and (H.3), respectively. To prove that scheme $(\mathbf{S}, \mathbf{R})$ is statistically hiding, it suffices to show that there exists sets $\Gamma_1', \Gamma_2' \subseteq \{0,1\}^{nr}$ such that the following holds for every adversarial receiver R^*:

(H'.1) Both $\mu(\Gamma_1'), \mu(\Gamma_2') \geq 1 - 2^{-\Omega(n)}$.

(H'.2) (A', V') is $2^{-\Omega(n)}$-close to (U_n, V'), where $A' = \mathrm{output}_{\mathbf{S}}(\mathbf{S}_c^1(\Gamma_1'), R^*)$ denotes the private output of the sender $\mathbf{S}$ in the first phase, and $V' = \mathrm{view}_{R^*}(\mathbf{S}_c^1(\Gamma_1'), R^*)$ denotes the view of the adversarial receiver R^* in the first phase.

(H'.3) For all $\tau' \in \mathrm{Supp}(\mathrm{T}')$, $(B'_{\tau'}, W'_{\tau'})$ is $2^{-\Omega(n)}$-close to $(U_n, W'_{\tau'})$, where random variable $(B'_{\tau'}, W'_{\tau'}) = (\mathrm{output}_{\mathbf{S}}(\mathbf{S}_c^2(\Gamma_2'), R^*), \mathrm{view}_{R^*}(\mathbf{S}_c^2(\Gamma_2'), R^*))|_{\mathrm{T}'=\tau'}$, and random variable $\mathrm{T}' = \mathrm{transcript}(\mathbf{S}^1(\Gamma_2'), R^*)$. In other words, $B'_{\tau'}$ is the private output of the sender $\mathbf{S}$ in the second phase given that the first-phase transcript is τ' and $W'_{\tau'}$ is the view of the adversarial receiver R^* in the second phase given that the first-phase transcript is τ'.

Property (H.1) implies (H'.1). Let Γ_1 and Γ_2 be the corresponding sets for (S, R), and let $p = \mu(\Gamma_1)$. Set $\beta = \lfloor pn - \frac{1}{2}\delta n\rfloor$, $\gamma_1 = \lfloor pn - \frac{1}{12}\delta n\rfloor$ and $\gamma_2 = \lfloor(1 - p + \delta)n - \frac{1}{12}\delta n\rfloor$. Note that $\beta \in [0, n]$ since $p \in [\delta, 1]$.

Define the sets Γ_1' and Γ_2' as follows:

$$\Gamma_1' = \{(x_1, \ldots, x_n) : \exists\ i_1, \ldots, i_{\gamma_1} \text{ such that } x_{i_1}, \ldots, x_{i_{\gamma_1}} \in \Gamma_1\},$$
$$\Gamma_2' = \{(x_1, \ldots, x_n) : \exists\ i_1, \ldots, i_{\gamma_2} \text{ such that } x_{i_1}, \ldots, x_{i_{\gamma_2}} \in \Gamma_2\}.$$

To lower bound $\mu(\Gamma_1')$, note that $\mu(\Gamma_1)-\gamma_1/n = p - \left\lfloor pn - \frac{1}{12}\delta n \right\rfloor / n \geq \frac{1}{12}\delta = \Omega(1)$ since $\delta = \Omega(1)$. Using a Chernoff bound, we get

$$\begin{aligned}\mu(\Gamma_1') &= 1 - \Pr_{(x_1,\ldots,x_n)}[\text{fewer than } \gamma_1 \text{ of the } x_i\text{'s are in } \Gamma_1] \\ &= 1 - 2^{-\Omega(n)}.\end{aligned}$$

To analyze $\mu(\Gamma_2')$, we note that $\mu(\Gamma_2)-\gamma_2/n \geq (1-p+\delta) - \left\lfloor (1-p+\delta)n - \frac{1}{12}\delta n \right\rfloor / n \geq \frac{1}{12}\delta = \Omega(1)$. Using a similar analysis as above, we get $\mu(\Gamma_2') = 1 - 2^{-\Omega(n)}$.

Property (H.2) implies (H'.2). Using the same notations and analysis as in the proof of Lemma 6.19, we let $A_i(x)$ denote the private output of the sender and $V_i(x)$ the view of the receiver in the i'th execution, for x being the private input for $\mathbf{S}_c^1$. That is, for $i = 1, \ldots, n$,

$$\begin{aligned}A_i(x) &= \mathrm{output}_S(S_c^1(x_i), R^*(V_1, \ldots, V_{i-1})); \\ V_i(x) &= \mathrm{view}_{R^*}(S_c^1(x_i), R^*(V_1, \ldots, V_{i-1})).\end{aligned}$$

Let $A''(\Gamma_1') = (A_1(\Gamma_1'), \ldots, A_n(\Gamma_1'))$ represent the combined first-phase private outputs of the n senders, and $V''(\Gamma_1') = (V_1(\Gamma_1'), \ldots, V_n(\Gamma_1'))$ represent the view of R^* when interacting with these n senders, before interactive hashing is done. From now on, we simplify notation by making $A'' = A''(\Gamma_1')$ and $V'' = V''(\Gamma_1')$.

Similar to (6.3) as in the proof of Lemma 6.19, we obtain

$$\mathrm{CP}^{1/2}(A''|V'') \leq \sqrt{2^{-\gamma_1 \cdot (k-1)}}\ .$$

And by a Markov bound, we know that with probability greater than $1 - 2^{-n}$ over $v'' \leftarrow V''$,

$$\mathrm{CP}(A''|_{V''=v''}) \leq 2^{-\gamma_1(k-1)} \cdot 2^{2n} \leq 2^{-\alpha_1 - (1/24)\delta k n + 3n} \leq 2^{-(\alpha_1+n)}, \qquad (6.5)$$

with the last inequality following from $k \geq 100/\delta$.

Consider $v'' \in V''$ such that the above (6.5) holds. Let $Q = (H_1, H_1(A''))$, where H_1 is an independent random hash from $\mathcal{H}_1$. Because H_1 is independent, $Q|_{V''=v''} = (H_1, H_1(A''|_{V''=v''}))$, and we can apply the Leftover Hash Lemma 6.8 to obtain that $Q|_{V''=v''}$, the input to the interactive hashing protocol, is $2^{-\Omega(n)}$-close to uniform.

Next, let $A' = \mathrm{output}_{\mathbf{S}}(\mathbf{S}^1(\Gamma_1'), R^*)$ denote the private output of $\mathbf{S}$ in the first phase, which in turn is equal to the output of S_{IH} in the interactive hashing protocol, so equivalently $A' = \mathrm{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q), R^*)$. Similarly, let $V' = \mathrm{view}_{R^*}(S_c^1(\Gamma_1), R^*)$ denote the view of the adversarial receiver R^* in the first phase, and let $V_{\mathrm{IH}} = \mathrm{view}_{R^*_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q), R^*_{\mathrm{IH}})$ denote the view of receiver R^* during the interactive hashing execution only. Observe that $V' = (V'', V_{\mathrm{IH}})$, recalling that V'' is the view of R^* when interacting with these n senders, before interactive hashing is done.

Because $Q|_{V''=v''}$, the input to interactive hashing, is $2^{-\Omega(n)}$-close to uniform, we know that $(A'|_{V''=v''}, V_{\mathrm{IH}}|_{V''=v''})$ is $2^{-\Omega(n)}$-close to $(U_n, V_{\mathrm{IH}}|_{V''=v''})$, as guaranteed by the hiding property of interactive hashing (see Definition 3.1). So $\mathbf{S}$'s private output $A'|_{V''=v''}$ is hidden from R^* for any $v'' \in V''$ satisfying the above (6.5). Finally note that (6.5) is satisfied for all but a 2^{-n} fraction of $v'' \leftarrow V''$, so it follows that (A', V') is $2^{-\Omega(n)}$-close to (U_n, V'), as required.

Property (H.3) implies (H'.3). Using similar ideas in the proof of Lemma 6.19, we can proceed as above and obtain that Property (H'.3) holds assuming (H.3). □

6.3.2. Binding preservation. In the execution of Algorithm 6.17, we obtained ℓ intermediate commitment schemes $[(S_j, R_j)]_{1\leq j\leq \ell}$, and one final commitment scheme (S, R). Our goal is to prove that the final scheme (S, R) satisfies the 1-out-of-2 binding property of Definition 5.4. To achieve our goal, we inductively show that the *expected* number of openings a sender can produce in the intermediate schemes is bounded by some constant, namely 32. (This is captured by Lemma 6.22 below.) Then in the final step, for scheme (S, R), we show how to shrink this expectation to value that is very close to 1, thereby proving that scheme (S, R) is satisfies the 1-out-of-2 binding property. (This in turn is captured by Lemma 6.24.)

In the definition of the computational 1-out-of-2 binding property (Definition 5.4), we stipulated that the adversarial sender in the second phase can be computationally unbounded, whereas the adversarial sender in the first phase must be probabilistic polynomial time (PPT). It will be rather messy to work with PPT senders, hence we will abstract away the PPT requirement by showing, in the next section, how to convert any PPT sender violating the 1-out-of-2 binding property in the first phase into a computationally unbounded sender with a special *unique binding* property. A sender with the unique binding property, intuitively, will not break the (first-phase) binding property of any execution of the initial schemes (S_0, R_0), but might still break the binding property of the intermediate schemes (S_j, R_j) (or final scheme (S, R)). Intuitively, we can restrict to such senders because of the computational 1-out-of-2 binding property of the initial scheme (S_0, R_0). Once we have a sender with the unique binding property, the analysis of the amplification steps is entirely information theoretic.

To formally define the unique binding property for senders, we observe that schemes $[(S_j, R_j)]_{1\leq j\leq \ell}$ and (S, R) each contain multiple executions of initial scheme (S_0, R_0). Hence, when a cheating sender S^* interacts with R_j, it is actually also interacting with the i-th execution of R_0, for each $i = 1, 2, \ldots$, which we will denote by $R_0[i]$. Formally, we define a (computationally unbounded) cheating sender strategy $S^*[i]$ that interacts with this single execution of $R_0[i]$ (more precisely, the first commit stage $R^1_{0,c}[i]$) by simulating all of the other messages of R_j on its own until the end of the first commit stage of $R_0[i]$. Then it enumerates over all choices for the subsequent messages of R_j and outputs all of the resulting full transcripts of S^*'s interactions with $R_0[i]$ (including both phases)

Let $\mathcal{B}^t_0$ be the binding set guaranteed by Lemma 6.14 for (S_0, R_0) with parameter t. Then we define the random variable openings$(S^*[i], R^1_0[i])(\mathcal{B}^t_0)$ as follows, analogously to Section 6.2.4:

- $S^*[i]$ and $R^1_{0,c}[i]$ interact to get first phase commitment $c^{(1)}$. (According to the definition of S^* above, $S^*[i]$ accomplishes this by simulating the rest of the interaction of S^* and R_j on its own.)
- After the interaction, $S^*[i]$ outputs a sequence of values $d^{(1)}_1, \ldots, d^{(1)}_\ell$ and corresponding full transcripts $\lambda_1, \ldots, \lambda_\ell$ of *both* phases. Recall that $\lambda_i = (\tau_i, \kappa_i)$, where τ_i and κ_i are the first-phase and second-phase transcripts, respectively. (According to the definition of S^* above, $S^*[i]$ accomplishes this by enumerating over all possible continuations of the interaction of S^* and R_j, and outputting all of the resulting transcripts of the interaction with $R_0[i]$.)
- We let openings$(S^*[i], R^1_0[i])(\mathcal{B}_0)$ be the set of distinct values $d^{(1)}_i$ whose opening λ_i is valid, where by valid we mean that λ_i begins with prefix $c^{(1)}$, λ_i contains a decommitment of $c^{(1)}$ to the value $d^{(1)}_i$ with first-phase transcript

$\tau_i \notin \mathcal{B}_0^t$, and both $R^1_{0,r}$ and $R^2_{0,r}$ accept in λ_i.

Definition 6.21. *(Unique binding property of sender.) For an intermediate scheme (S_j, R_j) or the final scheme (S, R) with parameter t, a (deterministic) sender S^* has the* **unique $\mathcal{B}_0^t$-binding** *property if for all i, we have*

$$|\operatorname{openings}(S^*[i], R_0^1[i])(\mathcal{B}_0^t)| \leq 1$$

with probability 1 (over the coins of $S^*[i]$[10] *and $R_0^1[i]$) where $S^*[i]$ and* $\operatorname{openings}(S^*[i], R_0^1[i])(\mathcal{B}_0^t)$ *are defined as above.*

The following two lemmas, Lemma 6.22 and 6.24, provide us a way to understand the binding property (in an average case sense) of $(\mathbf{S}, \mathbf{R})$, the amplified hiding scheme as presented in Protocol 6.16, in terms of (S, R). We might occasionally drop the superscript notations (1) and (2) from the notations if it is clear which phase we are referring to.

Lemma 6.22. *(Intermediate step binding preservation.) For every sufficiently large constant $D \in \mathbb{N}$, the following holds: For all integers $t \in [1, n]$, $j \in [1, \ell]$, $\beta_1, \ldots, \beta_j \in \{0, 1, \ldots, D-1\}$, letting (S_j, R_j) be the intermediate commitment schemes obtained in the execution of Algorithm 6.17 with parameters D, t, and $(\beta_1, \ldots, \beta_j)$,*[11] *there exists a binding set $\mathcal{B}_j = \mathcal{B}_j(t, \beta_1, \ldots, \beta_j)$ such that the following two conditions hold:*

(B.1) For every deterministic sender S^* with the unique $\mathcal{B}_0^t$-binding property*,*

$$\mathrm{E}\left[\left|\operatorname{openings}(S^*, R_j^1)(\mathcal{B}_j)\right|\right] < 32 \ ,$$

where the expectation is taken over the coin tosses of R_j^1.

(B.2) For every $\tau \in \mathcal{B}_j$ and for every deterministic sender S^*,

$$\mathrm{E}\left[\left|\operatorname{openings}(S^*, R_j^2)(\tau)\right|\right] < 32 \ ,$$

where the expectation is taken over the coin tosses of R_j^2.

Proof. We proceed to prove by induction on j. In fact, we will start with a base case of $j = 0$, i.e., consider the scheme (S_0, R_0) from Section 6.2. By Definition 6.21, the scheme (S_0, R_0) satisfies condition (B.1), and by Lemma 6.14, it satisfies condition (B.2).

For the inductive step, we assume (S_j, R_j) satisfy both (B.1) and (B.2), and show that so does (S_{j+1}, R_{j+1}). Note that (S_{j+1}, R_{j+1}) is obtained by the amplification procedure (Protocol 6.16) that combines m sequential executions of (S_j, R_j), i.e., $(S_{j+1}, R_{j+1}) = \mathsf{Amplify}(S_j, R_j)$. Hence, for convenience of notation we will denote (S_j, R_j) and (S_{j+1}, R_{j+1}) as (S, R) and $(\mathbf{S}, \mathbf{R})$ respectively. The i-th execution of (S, R) in $(\mathbf{S}, \mathbf{R})$ is denoted as $(S[i], R[i])$, not to be confused with the subscript indexing notation of (S_j, R_j).

Also throughout this proof, the value of m will be fixed to D, although we will keep writing m. Let $\mathcal{B} = \mathcal{B}_j$ be the binding set for (S, R). We define our new binding set $\mathcal{B}' = \mathcal{B}_{j+1}$ for $(\mathbf{S}, \mathbf{R})$ in terms of $\mathcal{B}$ and $\beta = \beta_{j+1}$ as follows:

$$\mathcal{B}' = \{(\tau_1, \ldots, \tau_m) : \exists\ j_1, \ldots, j_{\beta+1} \text{ such that } \tau_{j_1}, \ldots, \tau_{j_{\beta+1}} \in \mathcal{B}\} \ .$$

That is, a transcript $\tau' = (\tau_1, \ldots, \tau_m) \in \mathcal{B}'$ if and only if at least $\beta + 1$ of τ_j's are in $\mathcal{B}$. So, $\tau' \notin \mathcal{B}'$ if and only if at least $m - \beta$ of the τ_j's are not in $\mathcal{B}$.

[10] Note that $S^*[i]$ is probabilistic even if S^* is deterministic, because it simulates all of the random choices of R_j other than those of $R_0[i]$.

[11] Note that (S_j, R_j) does not depend on $\beta_{j+1}, \ldots, \beta_{\ell+1}$.

Property (B.1). Consider a deterministic S^* with the unique $\mathcal{B}_0^t$-binding property interacting with $\mathbf{R}^1$. The random coins of $\mathbf{R}^1$ can be broken up into independent random coins of $R^1[1], \ldots, R^1[m]$ and R^1_{IH}, the receiver in the interactive hashing.

Recall that the m executions of (S, R) in $(\mathbf{S}, \mathbf{R})$ are sequential. We want to focus on the interaction of S^* with (the commit phase of) $R^1[i]$. To do so, for each possible setting of the coin tosses $r_1, \ldots, r_{i-1}$ for $R^1[1], \ldots, R^1[i-1]$, define a sender strategy $S^*[[i]](r_1, \ldots, r_{i-1})$ that interacts with $R^1[i]$, as follows: $S^*[[i]](r_1, \ldots, r_{i-1})$ simulates S^* using fixed coins r_j for all the previous $R^1[j]$'s (for all $j < i$) and after the interaction with $R^1[i]$, $S^*[[i]]$ outputs all the valid openings that occur in some continuation of S^*'s interaction with $R[i]$ (by enumerating over all coins of the future $R[j]$'s, $j > i$, the coins of R^1_{IH}, and the coins of $\mathbf{R}^2$). Observe that $S^*[[i]](r_1, \ldots, r_{i-1})$ inherits the unique $\mathcal{B}_0^t$-binding property from S^*.

Let $X_i(r_1, \ldots, r_i) = \left|\mathrm{openings}(S^*[i](r_1, \ldots, r_{i-1}), R^1[i](r_i))(\mathcal{B})\right|$; this counts the number of valid decommitments in i-th execution, when the sender uses simulated coins $r_1, \ldots, r_{i-1}$ and $R^1[i]$ uses coins r_i. Let $U = (U_1, \ldots, U_m)$, where U_i denotes the uniform random variable on coins r_i for $R[i]$; note that these are independent because the honest receiver tosses independent coins for each execution. We now consider the random variables $X_i(U) = X_i(U_1, \ldots, U_i)$.

By our induction hypothesis applied to $S^*[[i]](r_1, \ldots, r_{i-1})$, for all fixed $(r_1, \ldots, r_{i-1})$, we have

$$\mathop{\mathrm{E}}_{U}[X_i(U) | U_1 = r_1, \ldots, U_{i-1} = r_{i-1}] = \mathop{\mathrm{E}}_{U_i}[X_i(r_1, \ldots, r_{i-1}, U_i)] < 32 \ .$$

Because the previous $X_j(U)$'s, for $j < i$, only depend on $U_1, \ldots, U_j$, we have that the expected value of X_i is less than 32 even given any previous values of X_j's. That is, $\mathrm{E}\left[X_i|_{X_1 = x_1, \ldots, X_{i-1} = x_{i-1}}\right] < 32$ for every $(x_1, \ldots, x_{i-1}) \in \mathrm{Supp}(X_1, \ldots, X_{i-1})$. The following claim allows us to bound the expectation of the product of these random variables.

Claim 6.23. *Let $Y_1, \ldots, Y_\ell$ be nonnegative real-valued random variables such that for all $i = 1, 2, \ldots, \ell$, we have* $\mathrm{E}[Y_i|_{Y_1 = y_1, \ldots, Y_{i-1} = y_{i-1}}] < \alpha_i \in \mathbb{R}^+$, *for every* $(y_1, \ldots, y_{i-1}) \in \mathrm{Supp}(Y_1, \ldots, Y_{i-1})$. *Then,*

$$\mathrm{E}\left[\prod_{i=1}^{\ell} Y_i\right] < \prod_{i=1}^{\ell} \alpha_i \ .$$

Proof. [Proof of Claim] Note that

$$\begin{aligned}\mathrm{E}[Y_1 \cdots Y_\ell] &= \mathop{\mathrm{E}}_{(y_1, \ldots, y_{\ell-1}) \xleftarrow{\mathrm{R}} (Y_1, \ldots, Y_{\ell-1})}\left[y_1 \cdots y_{\ell-1} \cdot \mathrm{E}[Y_\ell|_{Y_1 = y_1, \ldots, Y_{\ell-1} = y_{\ell-1}}]\right] \\ &< \mathrm{E}[Y_1 \cdots Y_{\ell-1} \cdot \alpha_\ell] \\ &= \alpha_\ell \cdot \mathrm{E}[Y_1 \cdots Y_{\ell-1}] \ ,\end{aligned}$$

and the claim follows by induction on ℓ. □

As noted above, it is always the case that $\mathrm{E}[X_i] < 32$, regardless of the instantiation of previous X_j's, for $j < i$. Note that Claim 6.23 also applies to computing the expectation of $\prod_{i \in J} X_i$, for any subset $J \subset [m]$, since any subset of the X_i's (preserving the right order) satisfy the condition of the claim.

Once the m commitments $R^1[i]$ are complete, we can define a random variable $A = A(U)$ that denotes the set of values $a = (a_1, \ldots, a_m)$'s for which the sender

S^* produces a valid opening with respect to $\mathcal{B}'$ in some continuation of the protocol when receiver $R^1[i]$ uses coin tosses U_i. By the definition of $\mathcal{B}'$, this means that $a = (a_1, \ldots, a_m)$ is valid only if at least $m - \beta$ of those are a_i's corresponding to decommitments that are not in $\mathcal{B}$. For those a_i's corresponding to decommitments that are in $\mathcal{B}$, the number of possible values that a_i can take on is $X_i(U)$. And for those a_i's corresponding to decommitments that are not in $\mathcal{B}$, we can bound the number of possible values that a_i can take on by 2^k (since a_i is a k-bit string).

$$\begin{aligned}
\mathop{\mathrm{E}}_{U}\left[|A(U)|\right] &\le \mathop{\mathrm{E}}_{U}\left[\sum_{J\subseteq[m],|J|\ge m-\beta}\prod_{i\in J}X_i(U)\prod_{i\notin J}2^k\right] \\
&= \sum_{J\subseteq[m],|J|\ge m-\beta}\mathop{\mathrm{E}}_{U}\left[\prod_{i\in J}X_i(U)\prod_{i\notin J}2^k\right] \\
&< \sum_{J\subseteq[m],|J|\ge m-\beta}\prod_{i\in J}32\cdot\prod_{i\notin J}2^k && \text{(by Claim 6.23)} \\
&\le 2^m\cdot 32^{m-\beta}\cdot(2^k)^{\beta} && \text{(because } 32<2^k\text{)} \\
&\le 2^{(\beta+1)(k-1)+6m-k+1} = 2^{\alpha_1-(k-6m-4)} \ .
\end{aligned}$$

Define random variable $T_1 = T_1(U) = \{(h_1, h_1(a)) : a \in A, h_1 \in \mathcal{H}_1\}$. Since $\mathrm{E}[|A|] \le 2^{\alpha_1-(k-6m-4)}$ and the range of $h_1 \in \mathcal{H}_1$ is of length α_1, the expected density of T_1 satisfies $\mathrm{E}[\mu(T_1)] \le \mathrm{E}[|A|]\cdot 2^{-\alpha_1} \le 2^{-(k-6m-4)}$, where the expectations are taken over the coin tosses $U = (U_1, \ldots, U_m)$. Note that T_1 is independent of the coins of R^1_{IH} in the first phase interactive hashing (though not independent of the coins of $\mathbf{R}^1$).

Finally, we have

$$\mathop{\mathrm{E}}_{\text{coins }\mathbf{R}^1}\left[\left|\mathrm{openings}(S^*,\mathbf{R}^1)(\mathcal{B}')\right|\right] \le \mathop{\mathrm{E}}_{T_1,\text{ coins }R^1_{\mathrm{IH}}}\left[\left|\{d^{(1)} : C^{(1)}(d^{(1)}) \in T_1\}\right|\right] \ ,$$

where in the second expectation, $C^{(1)} = \mathrm{output}(S^*, R^1_{\mathrm{IH}})$. By Lemma 6.10,

$$\mathop{\mathrm{E}}_{T_1,\text{ coins }R^1_{\mathrm{IH}}}\left[\left|\{d^{(1)} : C^{(1)}(d^{(1)}) \in T_1\}\right|\right] < 24 + 2^{k'+1}\cdot\mathrm{E}[\mu(T_1)] < 32 \ ,$$

with the last inequality following from $k' = k - 8m - 8$.

Property (B.2). We use the same approach as above, except this time, we consider all deterministic S^*, as opposed to only those with the unique binding property. Also we need to fix a binding transcript $\tau = (\tau_1, \ldots, \tau_m) \in \mathcal{B}'$. Let J be the set of indices such that $\tau_i \in \mathcal{B}$.

Analogously to the proof of Property (B.1), we define $S^*[[[i]]](r_1, \ldots, r_{i-1})$ for every possible setting $r_1, \ldots, r_{i-1}$ of coin tosses for receivers $R^2[1], \ldots, R^2[i-1]$, and we set $X_i(r_1, \ldots, r_i) = \left|\mathrm{openings}(S^*[[[i]]](r_1, \ldots, r_{i-1}), R^2[i](r_i))(\tau_i)\right|$. By our induction hypothesis, for all $i \in J$, we have

$$\mathrm{E}\left[X_i|_{X_1=x_1,\ldots,X_{i-1}=x_{i-1}}\right] < 32 \ ,$$

for any $(x_1, \ldots, x_{i-1}) \in \mathrm{Supp}(X_1, \ldots, X_{i-1})$, where we define the random variables $X_i = X_i(U_1, \ldots, U_i)$ and $U = (U_1, \ldots, U_m)$ with U_i being uniform over the coin tosses r_i for $R^2[i]$.

Let random variable $B = B(U)$ denote the set of values $b = (b_1, \ldots, b_m)$ for which the sender S^* produces a valid opening in some continuation of the protocol. Noting that X_i can be as large as 2^k for indices $i \notin J$, we have

$$\begin{aligned}
\mathrm{E}\left[|B|\right] &\leq \mathop{\mathrm{E}}_{U}\left[\prod_{i \in J} X_i \prod_{i \notin J} 2^k\right] \\
&< \prod_{i \in J} 32 \cdot \prod_{i \notin J} 2^k && \text{(by Claim 6.23)} \\
&\leq 32^{\beta+1} \cdot (2^k)^{m-\beta-1} && \text{(because } 32 < 2^k\text{)} \\
&\leq 2^{(m-\beta)(k-1)-(k-6m)} && \text{(because } m > 5\text{)} \\
&= 2^{\alpha_2-(k-6m-3)}.
\end{aligned}$$

Let random variable $T_2 = T_2(U) = \{(h_2, h_2(b)) : b \in B, h_2 \in \mathcal{H}_2\}$. Since $\mathrm{E}[|B|] \leq 2^{\alpha_2-(k-6m-3)}$ and the range of $h_2 \in \mathcal{H}_2$ is α_2, the expected density of T_2 satisfies $\mathrm{E}[\mu(T_2)] \leq \mathrm{E}[|B|] \cdot 2^{-\alpha_2} \leq 2^{-(k-6m-3)}$, where the expectations are taken over $U = (U_1, \ldots, U_m)$. Note that T_2 is independent of the coins of R^2_{IH} in the second phase interactive hashing (though not independent of the coins of $\mathbf{R}^2$). Finally, we have

$$\mathop{\mathrm{E}}_{\text{coins } \mathbf{R}^2}\left[\left|\text{openings}(S^*, \mathbf{R}^2)(\tau')\right|\right] \leq \mathop{\mathrm{E}}_{T_2,\text{ coins } R^2_{\mathrm{IH}}}\left[\left|\{d^{(2)} : C^{(2)}(d^{(2)}) \in T_2\}\right|\right] \ ,$$

where in the second expectation, $C^{(2)} = \text{openings}(S^*(T_2), R_{\mathrm{IH}})$. By Lemma 6.10,

$$\mathop{\mathrm{E}}_{\text{coins } R^2_{\mathrm{IH}}, T_2}\left[\left|\{d^{(2)} : C^{(2)}(d^{(2)}) \in T_2\}\right|\right] < 24 + 2^{k'+1} \cdot \mathrm{E}[\mu(T_2)] < 32 \ ,$$

with the last inequality following from $k' = k - 8m - 8$. □

LEMMA 6.24. *(Final step binding preservation.) For every sufficiently large constant $D \in \mathbb{N}$, the following holds: For all integers $t \in [1, n]$, $\beta_1, \ldots, \beta_\ell \in \{0, 1, \ldots, D-1\}$, and $\beta_{\ell+1} \in [0, n]$, letting (S, R) be the final output of Algorithm 6.17 with parameters D, t, and $(\beta_1, \ldots, \beta_{\ell+1})$, there exists a binding set $\mathcal{B}'$ such that the following two conditions hold:*

(B.1) For every deterministic sender S^* with the unique $\mathcal{B}^t_0$-binding property*, with probability* $1 - 2^{-\Omega(n)}$ *over the coins of* R^1,

$$\left|\text{openings}(S^*, \mathsf{R}^1)(\mathcal{B}')\right| \leq 1 \ .$$

(B.2) For every $\tau \in \mathcal{B}'$ and for every deterministic sender S^*, with probability $1 - 2^{-\Omega(n)}$ *over the coins of* R^2,

$$\left|\text{openings}(S^*, \mathsf{R}^2)(\tau)\right| \leq 1 \ .$$

Proof. From Lemma 6.22, we have scheme (S_ℓ, R_ℓ) with an associated binding set $\mathcal{B} = \mathcal{B}_\ell$ satisfying both conditions (B.1) and (B.2) in Lemma 6.22. Scheme $(\mathsf{S}, \mathsf{R}) = \mathsf{Amplify}(S_\ell, R_\ell)$, and hence we will need to show that the amplification boosts the binding by making sure both $\left|\text{openings}(S^*, \mathsf{R}^1)(\mathcal{B}')\right| \leq 1$ and $\left|\text{openings}(S^*, \mathsf{R}^2)(\tau)\right| \leq 1$ with probability $1 - 2^{-\Omega(n)}$.

Throughout this proof, the value of m will be fixed to n (as in Step 3 of Algorithm 6.17), although we will keep writing m. We define our new binding set $\mathcal{B}'$ for (S, R) in terms of $\mathcal{B}$ and $\beta = \beta_{\ell+1}$ as follows:

$$\mathcal{B}' = \{(\tau_1, \ldots, \tau_m) : \exists\, j_1, \ldots, j_{\beta+1} \text{ such that } \tau_{j_1}, \ldots, \tau_{j_{\beta+1}} \in \mathcal{B}\} \ .$$

That is, a transcript $\tau' = (\tau_1, \ldots, \tau_m) \in \mathcal{B}'$ if and only if at least $\beta + 1$ of the τ_j's are in $\mathcal{B}$. Thus, $\tau' \notin \mathcal{B}'$ if and only if at least $m - \beta$ of the τ_j's are not in $\mathcal{B}$.

Property (B.1). Using the same analysis and notations as in the proof of Lemma 6.22, we have that

$$\mathop{\mathrm{E}}_{U}\left[|A|\right] \leq 2^m \cdot 32^{m-\beta} \cdot (2^k)^\beta \leq 2^{\beta k + 6m} \quad ,$$

where A is the random variable denoting the set of values $a = (a_1, \ldots, a_m)$'s for which the sender S^* produces a valid opening with respect to $\mathcal{B}'$ in some continuation of the protocol and $U = (U_1, \ldots, U_m)$ denotes coin tosses for $R^1[1], \ldots, R^1[m]$.

Since $\delta = \Omega(1)$ and $k = k_\ell \geq \log n$, observe that $\alpha_1 = \left\lfloor (\beta + \frac{1}{3}\delta n)k \right\rfloor = \beta k + \omega(n)$, for large enough values of n. Let random variable $T_1 = \{(h_1, h_1(a)) : h_1 \in \mathcal{H}_1, a \in A\})$. Since the range of $h_1 \in \mathcal{H}_1$ is $\{0,1\}^{\alpha_1}$, the density of $\mathit{\Gamma}_1$ satisfies

$$\mathop{\mathrm{E}}_{U}\left[\mu(T_1)\right] \leq \mathrm{E}[|A|] \cdot 2^{-\alpha_1} < 2^{\beta k + 6m} \cdot 2^{-(\beta k + \omega(n))} = 2^{-\omega(n)} \quad ,$$

since $m = n$. Thus, with probability at least $1 - 2^{-n}$ over the coin tosses U of $R^1[1], \ldots, R^1[m]$, we have that

$$\mu(T_1) \leq 2^{-\omega(n)} \cdot 2^n \leq 2^{-2n} \quad .$$

By Lemma 3.7, we can conclude that for such a T_1 (with $\mu(T_1) \leq 2^{-2n}$),

$$\Pr_{\text{coins } R^1_{\mathrm{IH}}}\left[\left|\{d^{(1)} : C^{(1)}(d^{(1)}) \in T_1\}\right| > 1\right] \leq \mathrm{poly}(n) \cdot (2^{-2n} \cdot 2^{k'})^{1/2} = 2^{-\Omega(n)} \quad .$$

Finally, we have:

$$\begin{aligned}
&\Pr_{\text{coins } \mathbf{R}^1}\left[\left|\mathrm{openings}(S^*, \mathbf{R}^1)\right| > 1\right] \\
&\leq \Pr_{\text{coins } R^1_1, \cdots, R^1_m}\left[\mu(T_1) > 2^{-2n}\right] + \Pr_{\text{coins } R^1_{\mathrm{IH}}}\left[|\{d^{(1)} : C^{(1)}(d^{(1)}) \in T_1\}| > 1 \;\middle|\; \mu(T_1) \leq 2^{-2n}\right] \\
&= 2^{-\Omega(n)} \quad .
\end{aligned}$$

Property (B.2). Fix any $\tau' \in \mathcal{B}'$. Again, we use the same analysis and notations as in the proof of Lemma 6.22 to get:

$$\mathop{\mathrm{E}}_{U}\left[|B|\right] \leq 32^{\beta+1} \cdot (2^k)^{m-\beta-1} \leq 2^{(m-\beta)k + 5m} \quad ,$$

where B is the random variable denoting the set of values $b = (b_1, \ldots, b_m)$'s for which the sender S^* produces a valid opening in some continuation of the protocol and $U = (U_1, \ldots, U_m)$ denotes coin tosses for $R^2[1], \ldots, R^2[m]$.

Since $\delta = \Omega(1)$ and $k \geq \log n$, observe that $\alpha_2 = \left\lfloor (n - \beta + \frac{1}{3}\delta n)k \right\rfloor = (n-\beta)k + \omega(n)$, for large enough values of n. Let random variable $T_2 = \{(h_2, h_2(b)) : h_2 \in \mathcal{H}_2, b \in B\}$. Since the range of $h_2 \in \mathcal{H}_2$ is $\{0,1\}^{\alpha_2}$, the density of T_2 satisfies

$$\mathop{\mathrm{E}}_{U}\left[\mu(T_2)\right] \leq \mathrm{E}[|B|] \cdot 2^{-\alpha_2} < 2^{(m-\beta)k + 5m} \cdot 2^{-((n-\beta)k + \omega(n))} = 2^{-\omega(n)} \quad ,$$

since $m = n$. Thus, with probability at least $1 - 2^{-n}$ over the coin tosses U of $R^2[1], \ldots, R^2[m]$, we have that

$$\mu(T_2) \leq 2^{-\omega(n)} \cdot 2^n \leq 2^{-2n}.$$

By Lemma 3.7, we can conclude that for such a T_2 (with $\mu(T_2) \leq 2^{-2n}$),

$$\Pr_{\text{coins } R^2_{\text{IH}}}\left[\left|\{d^{(2)} : C^{(2)}(d^{(2)}) \in T_2\}\right| > 1\right] = 2^{-\Omega(n)} \ .$$

Finally, we have:

$$\begin{aligned}&\Pr_{\text{coins } \mathbf{R}^2}\left[\left|\text{openings}(S^*, \mathbf{R}^2)(\tau')\right| > 1\right]\\ &\leq \Pr_{\text{coins } R^2_1, \cdots, R^2_n}\left[\mu(T_2) > 2^{-2n}\right] + \Pr_{\text{coins } R^2_{\text{IH}}}\left[|\{d^{(2)} : C^{(2)}(d^{(2)}) \in T_2\}| \ \middle| \ \mu(T_2) \leq 2^{-2n}\right]\\ &= 2^{-\Omega(n)} \ .\end{aligned}$$

□

6.4. A collection of 1-out-of-2-binding commitments. In this section, we prove Theorem 6.1 restated below.

Restatement of Theorem 1 (6.1). *Given a one-way function $f : \{0,1\}^n \to \{0,1\}^n$, we can construct in time polynomial in n a collection of $m = \text{poly}(n)$ public-coin 2-phase commitment schemes $\mathcal{COM} = \{\mathsf{Com}_1, \cdots, \mathsf{Com}_m\}$ with message lengths $(k_1, k_2) = (n, n)$, such that:*

- *there exists an index $i \in \{1, 2, \ldots, m\}$ such that scheme Com_i is statistically hiding, and*
- *for every index $i \in \{1, 2, \ldots, m\}$, scheme Com_i is computationally 1-out-of-2 binding.*

6.4.1. Proof of Theorem 6.1. To obtain the desired collection of two-phase commitment schemes, we apply Algorithm 6.17 to the weakly hiding scheme (S_0, R_0), which can be constructed based on any one-way function $f : \{0,1\}^n \to \{0,1\}^n$. More precisely, we obtain a collection of commitments by enumerating over all the polynomially many choices of the integers $t \in \{\lceil 2\log n\rceil, 2, \ldots, n - \lceil 2\log n\rceil\}$, $\beta_1, \ldots, \beta_\ell \in \{0, 1, \ldots, D-1\}$, and $\beta_{\ell+1} \in \{0, 1, \ldots, n\}$. Note that the number of choices is $n \cdot D^\ell \cdot (n+1) = \text{poly}(n)$, as $D = O(1)$ and $\ell = \log n$. By Lemma 6.18, the resulting commitment schemes $\mathsf{Com}_1, \cdots, \mathsf{Com}_m$ all run in polynomial time. The hiding and binding properties of these schemes are given by Lemmas 6.25 and 6.26, which together establish Theorem 6.1.

Lemma 6.25. *For every sufficiently large constant $D \in N$, the following holds: Let $f : \{0,1\}^n \to \{0,1\}^n$ be any function (not necessarily one-way) such that $|f^{-1}(y)| \in [2n^2, 2^n/n^2]$ for every $y \in \{0,1\}^n$. Then there exist integers $t \in \{\lceil 2\log n\rceil, 2, \ldots, n - \lceil 2\log n\rceil\}$, $\beta_1, \ldots, \beta_\ell \in \{0, 1, \ldots, D-1\}$, and $\beta_{\ell+1} \in \{0, 1, \ldots, n\}$ such that the two-phase commitment scheme (S, R) produced by Algorithm 6.17 with parameters D, t, and $(\beta_1, \ldots, \beta_{\ell+1})$ is statistically hiding in the sense of Definition 5.3.*

As we observed earlier with Theorem 5.6, the condition on the preimage sizes of f can be easily made to hold by padding the input of f. Specifically, given any function $f : \{0,1\}^n \to \{0,1\}^n$, the function $f'(x, y, z) = (f(x), y)$ with $|x| = |y| = |z|$ has input length $n' = 3n$, all preimage sizes between $2^{n'/3}$ and $2^{2n'/3}$, and is one-way if f is.

Proof. We prove by induction on $j = 0, \ldots, \ell$ that there exist $t, \beta_1, \ldots, \beta_j$ such that (S_j, R_j) is δ_j-hiding in $\text{CP}^{1/2}$ measure for $\delta_j = \min\{2^j/n, 1/D\}$. The base case of $j = 0$ is simply Lemma 6.13. The induction step is provided by the Intermediate Step Hiding Amplification Lemma 6.19. Specifically, if by induction we choose $t, \beta_1, \ldots, \beta_{j-1}$ such that (S_{j-1}, R_{j-1}) is δ_{j-1}-hiding, the lemma tells us that there exists β_j such that (S_j, R_j) is δ'-hiding for $\delta' = \min\{2\delta_{j-1}, 1/D\} = \delta_j$.

Finally, we apply the Final Step Hiding Amplification Lemma 6.20 to obtain a $\beta_{\ell+1}$ such that (S,R) is statistically hiding. The lemma applies because $\delta_\ell = \min\{2^\ell/n, 1/D\} = \Omega(1)$ (recall that $\ell = \lceil \log n \rceil$ and $D = O(1)$), and $k_\ell = (16D)\cdot\ell - (8D+8)\cdot\ell > 100/\delta_\ell$ for sufficiently large n. □

Lemma 6.26. *For every sufficiently large constant D, the following holds: If $f : \{0,1\}^n \to \{0,1\}^n$ is one way, then for all integers $t \in \{\lceil 2\log n\rceil, 2, \ldots, n - \lceil 2\log n\rceil\}$, $\beta_1,\ldots,\beta_\ell \in \{0,1,\ldots,D-1\}$, and $\beta_{\ell+1} \in \{0,1,\ldots,n\}$, the two-phase commitment scheme (S,R) produced by Algorithm 6.17 with parameters D, t, and $(\beta_1,\ldots,\beta_{\ell+1})$ is computationally 1-out-of-2 binding in the sense of Definition 5.4. (Here the function f for which the scheme is based on needs to be hard to invert.)*

Proof. Let $\mathcal{B}'$ be the binding set given by Lemma 6.24. That lemma establishes that the two-phase commitment scheme (S,R) produced by Algorithm 6.17 satisfies the second condition of Definition 5.4 with respect to $\mathcal{B}'$. In addition, it also satisfies the first condition for all S^* with the *unique binding* property. Stated formally, for every deterministic (and computationally unbounded) S^* with the unique $\mathcal{B}_0^t$-binding property,

$$\Pr\left[\left|\mathrm{openings}(S^*, \mathsf{R}^1)(\mathcal{B}')\right| \le 1\right] = 1 - 2^{-\Omega(n)}, \tag{6.6}$$

where the probability is taken over the coins of R^1.

Thus, it suffices to prove is that any PPT S^* breaking the first condition of Definition 5.4 (with respect to $\mathcal{B}'$) with probability ε will either (i) yield a *PPT* $\hat{S}$ that violates the computational 1-out-of-2 binding property of (S_0, R_0) with probability at least $\varepsilon^{O(1)}/\operatorname{poly}(n)$, or (ii) yield a computationally unbounded $\hat{S}$ that has the unique $\mathcal{B}_0^t$-binding property and violates (6.6).

From now on, let ε be the probability that the S^* breaks the second condition of Definition 5.4 with respect to scheme (S,R). This probability is taken over the coin tosses of both the receiver R and the cheating sender S^*. We will write $S^*(r)$ to denote S^* with its coin tosses fixed to r. By the way we defined (S,R), it contains polynomially many executions of (S_0, R_0). Let $N = n \cdot D^\ell$ denote the number of such executions.

Let $\mathbf{z}$ denote a transcript of (S^*, R). Contained in $\mathbf{z}$ is also a first-phase commitment $z[i]$ for the i-th execution of R_0, denoted $R_0^1[i]$ (for all $i = 1, 2, \ldots, N$). Let $\hat{z}[i]$ be the partial transcript of $\mathbf{z}$ up to and including the first commit stage of $R_0[i]$. Note that $z[i]$ is a suffix of $\hat{z}[i]$, and $\hat{z}[i]$ is a prefix of $\mathbf{z}$.

For all indices $i \in [N]$, partial transcripts $\hat{z}[i]$ ending with the first commit stage of $R_0[i]$ and $d \in \{0,1\}^{k_0}$, and coin tosses r for S^*, define

$$p_{i,\hat{z}[i],d,r} = \Pr_{\mathbf{z}\leftarrow(S^*(r),\mathsf{R})}\left[\mathbf{z} \text{ contains a valid opening of } z[i] \text{ to value } d \,|\, \mathbf{z} \text{ begins with } \hat{z}[i]\right] \quad ,$$

where the probability is taken over the coin tosses of R (which includes coins of the subsequent executions $R_0[j]$, $j > i$, as well as coins for many executions of R_{IH}). As usual by a valid opening, we mean that S^* sends a full transcript $\lambda[i] = (\tau[i], \kappa[i])$ of both phases of its interaction with $R_0[i]$ such that $\lambda[i]$ begins with $z[i]$, $\lambda[i]$ contains a decommitment of $z[i]$ to value d with a first-phase transcript $\tau[i] \notin \mathcal{B}_0^t$ and both $R_0^1[i]$ and $R_0^2[i]$ accept in $\lambda[i]$.

Let $K = 2^{k_0}$, where k_0 is the message length in (S_0, R_0). We have two cases to consider.

Case 1. There exists an $i \in [N]$ such that with probability at least $\varepsilon/4NK$ over $\hat{z}[i]$ and r, there exists $d \neq d'$ with both $p_{i,\hat{z}[i],d,r}, p_{i,\hat{z}[i],d',r} > \varepsilon/4NK$.

In this case, we violate the computational 1-out-of-2 binding property of (S_0, R_0) by considering the following sender $\hat{S}$ interacting with $R_0[i]$.

1. Select a random $i \leftarrow [N]$ and coin tosses r for S^*.
2. Run $S^*(r)$ with R, simulating all of the messages of R internally except for those of $R_0[i]$. Halting after the first commit stage of $R_0[i]$, we obtain a partial transcript $\hat{z}[i]$. From $\hat{z}[i]$, we get $z[i]$, the first-phase commitment of $R_0[i]$.
3. Record the current state ψ of $S^*(r)$ and R.
4. Continue the execution of $S^*(r)$ with R from ψ twice using independent randomness for R.
5. Examine the transcripts to look for valid decommitments to two different values d, d'.

Because our goal is to violate the computational 1-out-of-2 binding property of (S_0, R_0), we succeed in the above algorithm if $d \neq d'$ and decommitments produced are valid. We calculate our success probability as follows: We guess correct index $i \in [N]$ with probability $1/N$. Given that we guess the correct i, we get the desired $\hat{z}[i]$ and r with probability at least $\varepsilon/4NK$. Now, when we do two independent continuations of $\hat{z}[i]$ we arrive at two different decommitted values with probability greater than $(\varepsilon/4NK)^2$. Consequently, we violate the computational 1-out-of-2 binding property of (S_0, R_0) (i.e., win the game in Condition 2 of Definition 5.4) with probability greater than

$$\frac{1}{N} \cdot \frac{\varepsilon}{4NK} \cdot \left(\frac{\varepsilon}{4NK}\right)^2 = \left(\frac{\varepsilon}{n}\right)^{O(1)} \quad ,$$

since $K = 2^{k_0} = 2^{O(\log n)} = \text{poly}(n)$ and $N = n \cdot D^{\ell} = n \cdot O(1)^{O(\log n)} = \text{poly}(n)$. This forces ε to be a negligible function.

Case 2. For all $i \in [N]$, it holds that with probability greater than $1 - \varepsilon/4NK$ over $\hat{z}[i]$ and r, there is at most one d such that $p_{i,\hat{z}[i],d,r} > \varepsilon/4NK$.

Define $d^*(\hat{z}[i], r)$ to be the value of d that maximizes $p_{i,\hat{z}[i],d,r}$. Taking a union bound over other possible values $d' \neq d^*(\hat{z}[i], r)$, we have that

$$\begin{aligned}
&\Pr_{r,\mathbf{z} \leftarrow (S^*(r),\mathsf{R})}[\text{in } \mathbf{z},\ S^*(r) \text{ opens some } z[i] \text{ to a value other than } d^*(\hat{z}[i], r)] \\
&\leq \sum_{i=1}^{N} \left(\frac{\varepsilon}{4NK} \cdot K + \Pr_{\hat{z}[i],r}\left[\text{exists more than one } d \text{ such that } p_{i,\hat{z}[i],d,r} > \frac{\varepsilon}{4NK}\right] \right) \\
&< N \cdot \left(\frac{\varepsilon}{4NK} \cdot K + \frac{\varepsilon}{4NK} \right) \\
&< \frac{\varepsilon}{2} \ .
\end{aligned}$$

Let $\hat{S}(r)$ be the adversary that mimics $S^*(r)$ except that it halts and fails if $S^*(r)$ attempts to open some $z[i]$ to a value other than $d^*(\hat{z}[i], r)$, for some $i \in [N]$ and $\hat{z}[i]$. By the above calculations, the final outcome of $(\hat{S}, \mathsf{R}^1)$ will only differ with the original final outcome of (S^*, R^1) with probability at most $\varepsilon/2$ over r and the coins of R^1. In addition, for each r, $\hat{S}(r)$ has the unique $\mathcal{B}_0^t$-binding property. By (6.6) above, for every r, $\left|\text{openings}(\hat{S}(r), \mathsf{R}^1)(\mathcal{B}')\right| > 1$ occurs with at most negligible probability over the coins of R^1. Hence, $\left|\text{openings}(S^*(r), \mathsf{R}^1)(\mathcal{B}')\right| > 1$ occurs with probability at most $\text{neg}(n) + \varepsilon/2$ over r and the coins of R^1. We started off assuming that S^* breaks property (B.1) of scheme (S, R) with probability at least ε, that is to say $\left|\text{openings}(S^*, \mathsf{R}^1)(\mathcal{B}_0^t)\right| > 1$ with probability at least ε. Thus $\varepsilon \leq \text{neg}(n) + \varepsilon/2$, which implies that $\varepsilon = \text{neg}(n)$. □

7. Standard Commitments from 1-out-of-2-Binding Commitments.

In this section we show how to transform any $\binom{2}{1}$-binding two-phase commitment scheme into a standard computationally binding commitment scheme, which is statistically hiding given that the two-phase commitment is. Applied to our result from the previous section, this gives a family of commitment schemes that are all computationally binding, and one of which is statistically hiding. In the next section, we show how to convert any such family into a *single* statistically hiding commitment scheme. We accomplish this using a novel application of a *universal one-way hash family*, whose existence can be based on any one-way function [7] (see also [29]). Thus, our transformation can be based on any one-way function.

7.1. Overview.

We would like to use a two-phase commitment schemes to construct a (standard) commitment scheme. A naive attempt to design the commitment scheme may go as follows: First, the sender commits to some random string x using the first-phase commit stage. Then, the receiver flips a coin $\mathit{phase} \in \{\mathsf{first}, \mathsf{second}\}$, if $\mathit{phase} = \mathsf{first}$ then the first-phase commitment is used as the commitment (e.g., the sender sends to the receiver the exclusive-or of its secret with x). Otherwise ($\mathit{phase} = \mathsf{second}$), the two parties execute the first phase reveal stage and if successful (i.e., the receiver does not reject), they use the second-phase commitment (invoked with the transcript of the first-phase as input) as the commitment.

The intuition is that since the commitment is $\binom{2}{1}$-binding, the sender cannot cheat in both phases together and thus the receiver would catch a cheating sender with probability half. The problem is, however, that the sender can decide in which commitment he likes to cheat *after* knowing the value of *phase*. Hence, the sender can cheat successfully in both cases without violating the $\binom{2}{1}$-binding of the underlying protocol.

Our additional idea is to use ***universal one-way hash functions*** (UOWHFs) in order to force the sender to decide in which phase it is about to cheat *before* knowing the value of *phase*. UOWHFs are a relaxation of collision-resistant hash functions that were defined by Naor and Yung [23] and shown to be constructible from any one-way function by Rompel [7].[12] A UOWHF is a family of compressing functions such that no efficient adversary can succeed in the following game with nonnegligible probability. The adversary should first announce a value x. Then, on a uniformly selected hash function f (given to the adversary *after* it announces x), it succeeds if it can find $x' \neq x$ such that $f(x') = f(x)$.

Our implementation is as follows: After the first-phase commit stage, the receiver selects a random (universal one-way) hash function f and the sender sends back $y = f(x)$. The protocol proceeds essentially as the naive protocol above, where any time the first-phase reveal stage is executed in the naive protocol revealing the value x' (either in the reveal stage for $\mathit{phase} = \mathit{first}$ or in the commit-stage for $\mathit{phase} = \mathsf{second}$), the receiver also verifies that $f(x') = y$.

Assuming the hash function f is sufficiently compressing, the string x remains quite unpredictable even though $f(x)$ is sent to R (in the new variant of the protocol).

[12] A version of Rompel's result [7] that holds also for uniform adversaries was recently written by Katz and Koo [29], also adding missing details and fixing some errors.

Thus, in the case that *phase* $=$ first, we can still use the "entropy" remaining in x to hide the sender's secret (assuming it is sufficiently shorter than $|x| - |f(x)|$). To show the statistical hiding in the complementary case when *phase* $=$ second, it is sufficient to note that sending $f(x)$, does not compromise the hiding property of the second-phase commitment. All in all, the protocol is statistically hiding for both choices of *phase* and thus it is statistically hiding.

To argue about the binding of the protocol, recall that the 1-out-of-2-binding property informally states that with high probability after the first-phase commit stage, there exists a *single* value $\widetilde{x}$ that allows the sender to cheat in the second-phase commitment. Now, if the sender sends y such that $f(\widetilde{x}) = y$, then in order to cheat in the case *phase* $=$ first, it will have to open the first-phase commitment to a value $x' \neq \widetilde{x}$ such that $f(x') = y = f(\widetilde{x})$. This would imply the breaking of the universal one-way hash function. On the other hand, if $f(\widetilde{x}) \neq y$, then in the case *phase* $=$ second the sender is forced to open the first-phase commitment to a value different than $\widetilde{x}$. This guarantees that the sender cannot cheat in the second-phase commitment and thus in this case our protocol is binding. In conclusion, since y is sent before *phase* is chosen, we are guaranteed that our protocol is weakly binding (since intuitively there always exists a choice of *phase* that prevent the sender from cheating). We complete the construction by amplifying the above protocol into a full-fledged statistically hiding commitment scheme using standard techniques.

7.2. The Transformation. We present the transformation algorithm using an arbitrary family of functions $\mathcal{F}$, and will only require $\mathcal{F}$ to be a universal one-way hash family when we want to prove the hiding and binding security properties.

ALGORITHM 7.1. *The transformation, denoted as 2-to-1-*Transform*.*

Input: *security parameter* 1^n*, two-phase commitment scheme* (S,R) *with message lengths* $(k_1,k_2) = (n,1)$*, and a family of functions* $\mathcal{F} = \bigcup_n \mathcal{F}_n = \{f\colon \{0,1\}^n \to \{0,1\}^m\}$.

Output: *Commitment scheme* $(\mathbb{S},\mathbb{R})$ *as described by Protocol 7.2.*

. . . . Hence, we write the commitment scheme obtained as $(\mathbb{S},\mathbb{R}) = $ 2-to-1-Transform$((\mathsf{S},\mathsf{R}),\mathcal{F})$.

PROTOCOL 7.2. *Standard commitment scheme* $(\mathbb{S},\mathbb{R})$ *from two-phase commitment scheme* (S,R).

Security parameter: 1^n*, given as common input to both* $\mathbb{S}$ *and* $\mathbb{R}$.

Sender's private input: *Bit* $b \in \{0,1\}$.

Commit stage:

1. $\mathbb{S}$ *selects a uniform* $\sigma \leftarrow \{0,1\}^n$.
2. $\mathbb{S}$ *and* $\mathbb{R}$ *engage in* $(\mathsf{S}_c^1(\sigma),\mathsf{R}_c^1)(1^n)$*, with* $\mathbb{S}$ *acting as* S_c^1 *and* $\mathbb{R}$ *acting as* R_c^1*. Let* $c^{(1)}$ *be the common output of* S_c^1 *and* R_c^1 *after the interaction.*
3. $\mathbb{R}$ *chooses* $f \leftarrow \mathcal{F}_n$ *and sends it to* $\mathbb{S}$.
4. $\mathbb{S}$ *sends* $y = f(\sigma)$ *to* $\mathbb{R}$.
5. $\mathbb{R}$ *flips a random coin, represented by phase* $\leftarrow \{1,2\}$*, and sends phase to* $\mathbb{S}$.

   If phase $= 1$*, then proceed as follows:*

   (a) $\mathbb{S}$ *selects a random hash* $h \leftarrow \mathcal{H}$*, where* $\mathcal{H}$ *is a family of pairwise-independent hash functions with domain* $\{0,1\}^n$ *and range* $\{0,1\}$*, and sends* $(h, b \oplus h(\sigma))$ *to* $\mathbb{R}$.

   (b) $\mathbb{S}$ *and* $\mathbb{R}$ *both output* $(c^{(1)}, f, y, phase = 1, h, b \oplus h(\sigma))$ *as the commitment.*

   If phase $= 2$*, then proceed as follows:*

(a) *$\mathbb{S}$ runs S_r^1 to obtain the decommitment message $\gamma^{(1)}$ and first-phase transcript τ corresponding to both σ and $c^{(1)}$. $\mathbb{S}$ sends $(\sigma, \gamma^{(1)}, \tau)$ to $\mathbb{R}$.*

(b) *$\mathbb{S}$ and $\mathbb{R}$ engage in $(\mathsf{S}_c^2(b), \mathsf{R}_c^2)(1^n, \tau)$, with $\mathbb{S}$ acting as S_c^2 and $\mathbb{R}$ acting as R_c^2. Let $c^{(2)}$ be the common output of S_c^2 and R_c^2 after the interaction.*

(c) *$\mathbb{S}$ and $\mathbb{R}$ both output $(c^{(1)}, f, y, phase = 2, c^{(2)})$ as the commitment.*

Reveal stage:

To decommit to bit b, do the following depending the value of phase.

If phase $= 1$, then:

1. *$\mathbb{S}$ sends (b, σ) to $\mathbb{R}$;*
2. *If $y = f(\sigma)$ and the last component of the commitment equals $b \oplus h(\sigma)$, then* $\mathbb{R}$ accepts. *Otherwise,* $\mathbb{R}$ rejects.

If phase $= 2$, then:

1. *$\mathbb{S}$ runs S_r^2 to obtain the decommitment message $\gamma^{(2)}$, and sends $(b, \gamma^{(2)})$ to $\mathbb{R}$;*
2. *If $y = f(\sigma)$ and both R_r^1 and R_r^2 accept $(c^{(1)}, \sigma, \gamma^{(1)})$ and $(c^{(2)}, b, \gamma^{(2)})$, respectively, then* $\mathbb{R}$ accepts. *Otherwise,* $\mathbb{R}$ rejects.

7.3. Analyzing the Transformation. The hiding and binding security properties of Protocol 7.2 will rely on properties of $\mathcal{F}$ being a universal one-way hash family.

Our plan for the remaining of this section is as follows: (i) we present the definition of a universal one-way hash family due to Naor and Yung [23]; (ii) we separate the properties of a universal one-way hash family into two parts; and finally, (iii) we prove the hiding and binding properties of Protocol 7.2 based on these two separate properties.

Universal one-way hash family. In order to define a universal one-way hash family, we need to understand what it means for a family of functions to be *polynomial-time computable*.

DEFINITION 7.3. *A family of functions $\mathcal{F} = \bigcup_n \mathcal{F}_n = \{f \colon \{0,1\}^n \to \{0,1\}^m\}$ is* **polynomial-time computable** *if*

- *Every function $f \in \mathcal{F}_n$ is described by a bitstring of length $p(n)$ for some polynomial p. By abuse of notation, we also denote this description by f, and write $f \overset{R}{\leftarrow} \mathcal{F}_n$ to mean that it is chosen uniformly at random in $\{0,1\}^{p(n)}$. (A more general definition would allow the description of the function to be selected according to any polynomial-time samplable distribution, even one that requires private coin tosses. However, our stronger 'public-coin' definition is achieved by existing constructions, and can be useful in applications, such as constructing public-coin zero-knowledge arguments.)*
- *There exists a deterministic polynomial-time algorithm F such that for every n and every $f \in \mathcal{F}_n$, given the description of the function f and a string $x \in \{0,1\}^n$, F outputs the value of $f(x)$.*

DEFINITION 7.4. *A polynomial-time computable family of functions $\mathcal{F} = \bigcup_n \mathcal{F}_n = \{f \colon \{0,1\}^n \to \{0,1\}^m\}$ is a* **universal one-way hash family** *if $m < n$ and for all PPT A the following is negligible in n*

$$\Pr[(x, \mathsf{state}) \leftarrow A(1^n), f \leftarrow \mathcal{F}_n, x' \leftarrow A(x, \mathsf{state}, f) : x' \neq x \bigwedge f(x') = f(x)].$$

Remark 7.5.

- *In the above definition, we allow the adversary to transfer additional information, i.e.,* state*, between the selection of* x *and finding the collision. This* state *variable does not appear in the definition in Katz and Koo [29], which is otherwise identical to the above. However, any universal one-way hash family* $\mathcal{F}$ *meeting their weaker definition can be converted into one meeting the above definition by selecting* $f \overset{R}{\leftarrow} \mathcal{F}$, $s \overset{R}{\leftarrow} \{0,1\}^n$*, and defining* $f'(x) = f(x \oplus s)$*. (Intuitively, the random shift* s *turns an arbitrary point* x *selected by the adversary into a uniformly random point out of the adversary's control.)*
  The original definition of Naor and Yung [23] (also used by Rompel [7]) does not involve the adversary before f *is chosen at all, but rather requires that for* all $x \in \{0,1\}^n$, $A(x,f)$ *has a low probability of producing a collision (over the choice of* f *and* A*'s coin tosses). Their definition is suited for the case of nonuniform security (as the arbitrary* x *can be viewed as nonuniform advice), in which case it becomes equivalent to ours (since* A *can also have* state *hardwired nonuniformly).*
- *Although it is more natural for the security be parameterized in terms of the output length, namely* m*, our applications do not require hash functions that are shrinking by more than a polynomial factor. Hence for this reason, and in part for consistency, we keep* n *as our security parameter.*
- *Naor and Yung [23] showed that starting with a universal one-way hash family that is compressing by only one bit, namely* $m = n-1$*, more compression can be achieved, say* $m \leq n/2$*, by iterative application several hash functions chosen from the family. Moreover, it is easy to verify that the same construction holds also w.r.t. to Definition 7.4. Hence, without loss of generality, we can assume that our universal one-way hash family will has the feature that* $m \leq n/2$*.*

Two properties of a universal one-way hash family. A universal one-way hash family satisfying Definition 7.4 has the following two main properties.

Large preimages: most of the preimages have a large size. This follows from the compressing nature of hash functions: the output length m is much shorter than the input length n. (Recall that we can get a universal one-way hash family with $m \leq n/2$.) We formalize this property in Definition 7.6.

Target collision resistance: it is hard to find collisions with a value x announced *before* the hash function is given. We formalize this property in Definition 7.7

Definition 7.6. *A family of functions* $\mathcal{F} = \bigcup_n \mathcal{F}_n = \{f \colon \{0,1\}^n \to \{0,1\}^m\}$ *has the* **large preimages** *property if for every* $f \in \mathcal{F}$*, most elements in the range of* f *have large preimage sizes. Stated precisely, there exists a function* $\alpha(n) = \omega(1)$ *and a negligible function* ε*, such that for all values of* n*, the following holds:*

$$\Pr_{x \leftarrow \{0,1\}^n}\left[\left|f^{-1}(f(x))\right| \geq n^{\alpha(n)}\right] \geq 1 - \varepsilon(n) \ ,$$

for every function $f \in \mathcal{F}_n$*.*

Definition 7.7. *A family of functions* $\mathcal{F} = \bigcup_n \mathcal{F}_n = \{f \colon \{0,1\}^n \to \{0,1\}^m\}$ *has the* **statistical [resp., computational] target collision resistance** *property if for every [resp., every PPT]* A*, the following is negligible in* n*:*

$$\Pr[(x, \mathsf{state}) \leftarrow A(1^n), f \leftarrow \mathcal{F}_n, x' \leftarrow A(x, \mathsf{state}, f) : x' \neq x \bigwedge f(x') = f(x)] \ .$$

Remark 7.8. *In this paper we are only using families of function that are computational target collision resistance. Yet whenever possible we state the results also w.r.t. families with statistical target collision resistance, because this generalization has proved useful in subsequent work [39].*

Large preimages and target collision resistance are opposing properties. Specifically, it is impossible for a *single* family of functions to have large preimages and have *statistical* target collision resistance. The power of a universal one-way hash family comes from the fact that it has the large preimages property and has *computational* target collision resistance.

Lemma 7.9. *If $\mathcal{F} = \bigcup_n \mathcal{F}_n = \{f \colon \{0,1\}^n \to \{0,1\}^m\}$, for $m \leq n/2$, is a universal one-way hash family, then $\mathcal{F}$ has both the large preimages and the* computational *target collision resistance properties.*

Proof. The computational target collision resistance property follow directly from Definition 7.4. Hence, all we need to show is that the compressing nature of $\mathcal{F}$, when $m \leq n/2$, implies the large preimages property.

Fix $f \in \mathcal{F}_n$ and group the elements with small preimages into a set $S = \{y \in \{0,1\}^m : \left|f^{-1}(y)\right| < 2^{\frac{3}{4}n-m}\}$. Since $m \leq n/2$, every element $y \notin S$ has a preimage of size $\left|f^{-1}(y)\right| \geq 2^{\frac{3}{4}n-m} \geq 2^{n/4} = n^{\omega(1)}$. To complete the proof, we bound the probability of landing in S, which we do by a union bound over the elements in S (for which, there are at most 2^m):

$$\Pr_{x \leftarrow \{0,1\}^n}[f(x) \in S] = \Pr\left[\exists y \in S \text{ with } f(U_n) = y\right] < \frac{2^{\frac{3}{4}n-m}}{2^n} \cdot 2^m = 2^{-n/4} = \operatorname{neg}(n) \ .$$

□

Hiding. Having separated the properties of a universal one-way hash family into having large preimages and having target collision resistance, we now show that the large preimages property of $\mathcal{F}$ translates to the hiding property of the commitment scheme $(\mathbb{S}, \mathbb{R}) = \text{2-to-1-}\mathsf{Transform}((\mathsf{S}, \mathsf{R}), \mathcal{F})$.

Lemma 7.10. *If the family of functions $\mathcal{F}$ has the large preimages property, and the two-phase commitment scheme (S, R) is statistically hiding, then scheme $(\mathbb{S}, \mathbb{R}) =$ 2-to-1-$\mathsf{Transform}((\mathsf{S}, \mathsf{R}), \mathcal{F})$ is statistically hiding.*

Proof. What we need to show is that for any adversarial receiver R^*, the views of R^* in $(\mathbb{S}(0), R^*)$ and $(\mathbb{S}(1), R^*)$ are statistically indistinguishable. (In this proof, we drop the security parametrization of 1^n because it is clear from context.) We can, without loss of generality, only consider deterministic R^* because we can fix the adversary's coin tosses to maximize its distinguishing advantage. In the rest of this proof, we use *indistinguishability* and *hiding* to mean those of the statistical variant.

Let P denote the value of *phase* sent by R^*, and we break our hiding analysis to cases when $P = 1$ and $P = 2$. To formalize this case analysis, we say that random variables X and Y are ***indistinguishable on event*** E if for all D, $|\Pr[D(X) = 1 \wedge E] - \Pr[D(Y) = 1 \wedge E]|$ is negligible (in the security parameter n). What we will show is that the random variables $\text{view}_{R^*}(\mathbb{S}(0), R^*)$ and $\text{view}_{R^*}(\mathbb{S}(1), R^*)$ are indistinguishable on both events $P = 1$ and $P = 2$, thus allowing us to conclude that the scheme is hiding.

First, we analyze the case when $P = 2$. Let the random variables Σ and F denote $\mathbb{S}$'s choice of σ and the value of f sent by R^*, respectively. Observe that P is a *deterministic* function of the random variables $V_1 = \text{view}_{R^*}(\mathsf{S}^1_c(\Sigma), R^*)$ and $Y = F(\Sigma)$. In turn, V_1 and Y are deterministic functions of the first-phase transcript $\mathrm{T} = \text{transcript}(\mathsf{S}^1(\Sigma), R^*)$, which includes both the commit and reveal stages. This is because we can compute the view of the receiver from the first-phase transcript, and the first-phase transcript also contains the value of σ, from which we can compute $y = f(\sigma)$. For bit $b \in \{0,1\}$, let random variable $V_2(b) = \text{view}_{R^*}(\mathsf{S}^2_c(b), R^*)(\mathrm{T})$, recalling that $\mathrm{T} = \text{transcript}(\mathsf{S}^1(\Sigma), R^*)$. Because (S, R) is hiding, its second-phase commitment is hiding even given the first-phase transcript: this means that $(V_2(0), \mathrm{T})$ is indistinguishable from $(V_2(1), \mathrm{T})$. Since P is a deterministic function of T, random variables $(V_2(0), \mathrm{T})$ and $(V_2(1), \mathrm{T})$ are indistinguishable on event $P = 2$. Finally, since $\text{view}_{R^*}(\mathbb{S}(b), R^*)|_{P=2}$ is a deterministic function of $(V_2(b), \mathrm{T})|_{P=2}$, for $b \in \{0,1\}$, we have that $\text{view}_{R^*}(\mathbb{S}(0), R^*)$ and $\text{view}_{R^*}(\mathbb{S}(1), R^*)$ are indistinguishable on event $P = 2$.

Next, we analyze the case when $P = 1$. The hiding property of the first phase gives us

$$(V_1, \Sigma) \approx_s (V_1, U_n) \ ,$$

where U_n represent a uniform random variable over $\{0,1\}^n$, and is independent from V_1 and Σ. Recall that the random variable F denotes the function f sent by R^*. Since F is a deterministic function of V_1, we get

$$(V_1, F, F(\Sigma), \Sigma) \approx_s (V_1, F, F(U_n), U_n) \ .$$

Now, let the random variable H represent the hash function h selected by $\mathbb{S}$ when *phase* $= 1$. Note that H is independent of V_1, F, Σ, and U_n, so

$$(V_1, F, Y, H, H(\Sigma)) \approx_s (V_1, F, F(U_n), H, H(U_n)) \ , \tag{7.1}$$

recalling that $Y = F(\Sigma)$.

What we need to establish is that $H(U_n)$ is close to uniform so that we have hiding. The next claim does this for us.

Claim 7.11. *Suppose family of functions $\mathcal{F} = \bigcup_n \mathcal{F}_n$ has the large preimages property. Let the random variable H denote a random hash function from a family of pairwise-independent hash functions with domain $\{0,1\}^n$ and range $\{0,1\}$, random variable U_n denote a uniform string in $\{0,1\}^n$, random variable U_1' denote a uniform string in $\{0,1\}$, and that H, U_n, and U_1' are all independent. For every $f \in \mathcal{F}_n$, $(f(U_n), H, H(U_n))$ is indistinguishable from $(f(U_n), H, U_1')$.*

Proof. [Proof of Claim] The large preimages property of $\mathcal{F}$ guarantees that with probability $1 - \text{neg}(n)$ over $y \leftarrow f(U_n)$, the min-entropy $\mathrm{H}_\infty(U_n|_{f(U_n)=y}) \geq \omega(\log n)$. For y satisfying this condition, we apply the Leftover Hash Lemma 4.1 to get that $(y, H, H(U_n|_{f(U_n)=y}))$ is indistinguishable from $(y, H, H(U_n|_{f(U_n)=y}))$. □

Because H and U_n are independent from the rest of the random variables (and are independent from each other), Claim 7.11 states that

$$(V_1, F, F(U_n), H, H(U_n)) \approx_s (V_1, F, F(U_n), H, U_1') \ , \tag{7.2}$$

where U_1' is an independent random variable representing a uniform random variable over $\{0,1\}$. Combining (7.1) and (7.2), we get

$$(V_1, F, Y, H, H(\Sigma)) \approx_s (V_1, F, F(U_n), H, U_1') \ ,$$

which leads to:

$$\begin{aligned}(V_1, F, Y, H, 0 \oplus H(\Sigma)) &\approx_s (V_1, F, F(U_n), H, 0 \oplus U_1') \\ &\equiv\ (V_1, F, F(U_n), H, 1 \oplus U_1') \\ &\approx_s (V_1, F, Y, H, 1 \oplus H(\Sigma))\ .\end{aligned}$$

Since P is a deterministic function of V_1 and Y, random variables $(V_1, F, Y, H, 0 \oplus H(\Sigma))$ and $(V_1, F, Y, H, 1 \oplus H(\Sigma))$ are indistinguishable on event $P = 1$. Since $\mathrm{view}_{R^*}(\mathbb{S}(b), R^*)|_{P=1}$ is a deterministic function of $(V_1, F, Y, H, b \oplus H(\Sigma))|_{P=1}$, for $b \in \{0, 1\}$, we have that $\mathrm{view}_{R^*}(\mathbb{S}(0), R^*)$ and $\mathrm{view}_{R^*}(\mathbb{S}(1), R^*)$ are indistinguishable on event $P = 1$. □

Binding. We show that the target collision resistance property of $\mathcal{F}$ translates to the binding property of the commitment scheme $(\mathbb{S}, \mathbb{R}) = \text{2-to-1-}\mathsf{Transform}((\mathsf{S}, \mathsf{R}), \mathcal{F})$ obtained from the 2-to-1-$\mathsf{Transform}$. Because we will only be able to show that $(\mathbb{S}, \mathbb{R})$ is binding with probability close to $1/2$, we first define what it means to for a scheme to be binding with probability δ, for some $\delta \in [0, 1]$.

Definition 7.12. *Commitment scheme (S, R) is* **statistically [resp. computationally]** $\delta(n)$**-binding** *if for every [resp. every PPT] S^* and every large enough values of n, sender S^* succeeds in the following game with probability at most $\delta(n)$:*

> *On security parameter 1^n, S^* interacts with R in the commit stage obtaining commitment c. Then S^* outputs pairs $(0, d_0)$ and $(1, d_1)$, and* succeeds *if in the reveal stage, $R(0, d_0, c) = R(1, d_1, c) = \mathtt{accept}$.*

The standard notion of binding as given in Definition 2.4 corresponds to being computationally $1/p(n)$-binding for every polynomial p.

Lemma 7.13. *If the family of functions $\mathcal{F}$ is statistically [resp., computationally] target collision resistant, and the two-phase commitment scheme (S, R) is statistically [resp., computationally] $\binom{2}{1}$-binding, then the scheme $(\mathbb{S}, \mathbb{R}) =$ 2-to-1-$\mathsf{Transform}((\mathsf{S}, \mathsf{R}), \mathcal{F})$ is statistically [resp., computationally] $(1/2 + 1/p(n))$-binding for every polynomial p and sufficiently large n.*

Proof. We will focus on the case of computational binding. The statistical case will follow from the fact that the proof is "black box". Specifically, our proof will (implicitly) give efficient reductions M_1, M_2 such that given any sender strategy S^* that breaks the $(1/2 + 1/p(n))$-binding property of $(\mathbb{S}, \mathbb{R})$ as oracle, either $M_1^{S^*}$ will break the target collision resistance property of $\mathcal{F}$ with nonnegligible probability or $M_2^{S^*}$ will break the $\binom{2}{1}$-binding property of (S, R). If both $\mathcal{F}$ and (S, R) have statistical [resp., computational] security, then this is impossible for every strategy [resp., every PPT strategy] S^* and we deduce that $(\mathbb{S}, \mathbb{R})$ must be statistically [resp., computationally] $(1/2 + 1/p(n))$-binding.

Unless stated otherwise, we take probabilities over the entire interaction between S^* and $\mathbb{R}$ in both the commit and reveal stages. We say that S^* ***succeeds*** if it is able to produce decommitments to two different messages for commitment Υ in the reveal phase (recall that, the reveal stage is non-interactive). We want to prove that $\Pr[S^* \text{ succeeds}] \leq 1/2 + 1/p(n)$. We will do this by breaking the probability space into events $E_1, \ldots, E_5$ corresponding to the various cases in the intuitive proof outline given in Section 7.1. We will show that $\Pr[\bigvee_i E_i] = 1$, $\Pr[E_1] = 1/2$ and $\Pr[S^* \text{ succeeds} \wedge E_i] \leq 1/4p(n)$ for $i = 2, \ldots, 5$, and this will suffice to prove the lemma.

The first event, E_1, will depend on the random variables $C = \mathrm{view}_{S^*}(S^*, \mathsf{R}_c^1)$, representing S^*'s view of the first phase commit (this determines the entire state of the interaction (S^*, R), since by Definition 5.1 the honest receiver maintains no private state after the commit phase other than the commitment string); Y, denoting the hash value sent by S^* after the first-phase commit; P, representing the value of *phase*; and F, representing the choice of the function $f \overset{\text{R}}{\leftarrow} \mathcal{F}$. We would also like to consider whether or not Y equals $f(\Sigma^*)$, where Σ^* intuitively represents the value to which C is a commitment, i.e. the 'unique' value that will enable S^* to break the binding property of the 2nd phase. However, since the commitment scheme may be only computationally binding, Σ^* is not defined information-theoretically. Thus, we define it as the most likely value to which S^* will open the first-phase commitment (with a transcript not in $\mathcal{B}$). Formally, for each first-phase commit transcript $c \in \text{Supp}(C)$, we define:

$$p[\sigma \mid c] = \Pr\left[\begin{array}{c} (S^*, R) \text{ outputs an } \textit{accepting} \text{ full transcript } \lambda = (\tau, \kappa) \\ \text{such that } \tau \notin \mathcal{B} \text{ and } \tau \text{ contains an opening to } \sigma \end{array} | C = c \right] , \tag{7.3}$$

where the probability is over the random-coins of S^* and R, and we say full transcript λ is ***accepting*** if both R_r^1 and R_r^2 accept in λ. With this measure, we define $\sigma^*[c] = \text{argmax}_\sigma\, p[\sigma \mid c]$, breaking ties arbitrarily (say, by choosing the lexicographic smallest σ). Then we define the random variable $\Sigma^* = \sigma^*[C]$.

The intuition described in Section 7.1 suggests a case analysis based on whether or not $Y = F(\Sigma^*)$. According to that intuition, the scheme will be binding if $Y = F(\Sigma^*)$ and $P = 1$ (by target collision resistance of $\mathcal{F}$) or if $Y \neq F(\Sigma^*)$ and $P = 2$ (by the 1-out-of-2 binding property), and these events happen with probability 1/2 (because P is randomly chosen after Σ^*, F, and Y are determined). This intuition can be turned directly into a proof in the case that $\mathcal{F}$ has *nonuniform* target collision resistance, since the value of Σ^* (which is determined before F) can be hardwired into the adversary breaking $\mathcal{F}$. However, to prove our result for uniform adversaries as claimed, we need to ensure that $\Sigma^* = \sigma^*[C]$ can be efficiently computed (before being given F, as per Definition 7.7). We observe that this is the case if $p[\Sigma^* \mid C] > 1/4p(n)$, because then if we simulate a continuation of the execution of (S^*, R) starting after C, we have a non-negligible probability of Σ^* being revealed. On the other hand the case that $p[\Sigma^* \mid C] \leq 1/4p(n)$ turns out to be analyzable similarly to the case that $Y \neq F(\Sigma^*)$; in both cases we simply use the fact that S^* is unlikely to produce a successful opening to Σ^*.

With the above in mind, we begin by analyzing the event in which we do not expect the scheme to be binding.

Claim 7.14. *For the event*

$$E_1 = \left\{ \begin{array}{c} [(Y = F(\Sigma^*)) \wedge (p[\Sigma^* \mid C] > 1/4p(n))] \wedge [P = 2] \\ \vee[(Y \neq F(\Sigma^*)) \vee (p[\Sigma^* \mid C] \leq 1/4p(n))] \wedge [P = 1] \end{array} \right\},$$

we have $\Pr[E_1] = 1/2$.

Proof. [Proof of Claim] P is chosen randomly in $\{1, 2\}$ after C, Σ^*, F, and Y are determined. □

Now we want to show that the scheme is binding on the complement of E_1. First we handle the case that $P = 1$.

Claim 7.15. *For the event*

$$E_2 = \{[Y = F(\Sigma^*)] \wedge [p[\Sigma^* \mid C] > 1/4p(n)] \wedge [P = 1]\},$$

we have $\Pr[S^* \textit{ succeeds} \wedge E_2] \leq 1/4p(n)$.

Proof. [Proof of Claim] Suppose for contradiction that $\Pr[S^* \text{ succeeds} \wedge E_2] > 1/4p(n)$; we will show that we can break the target collision resistance property of

$\mathcal{F}$ with nonnegligible probability. In order to do so, we need to output an element x before seeing the hash function, and then given a random function $f \overset{\mathrm{R}}{\leftarrow} \mathcal{F}$, we need to output $x' \neq x$ such that $f(x) = f(x')$. We do this as follows. First we simulate the interaction between S^* and R up to the end of the first-phase commitment, and record c as the sender's view so far. Then we continue the interaction from c to the end and set x to be the value of σ sent by S^* in the protocol (if no valid value is given, we set x to some default value). (In case $\mathit{phase} = 1$ and S^* produces two values for σ in breaking the scheme, choose one of the two at random.) Now we output x and store $\mathsf{state} = c$, and receive a random hash function $f \overset{\mathrm{R}}{\leftarrow} \mathcal{F}$. We now rerun the interaction between S^* and R, starting with the view (c, f), and set x' to be the value of σ sent by S^* in the protocol (again choosing randomly if $\mathit{phase} = 1$ and S^* produces two values).

To see that this strategy breaks the target collision resistance property with nonnegligible probability, consider the second completed execution of the interaction between S^* and R (the one with the given, random, hash function f, which we now denote as a random variable F). By assumption, with probability greater than $1/4p(n)$ in this execution, it holds that S^* succeeds, $Y = F(\Sigma^*)$, $p[\Sigma^* \mid C] > 1/4p(n)$, and $P = 1$. Since S^* succeeds and $P = 1$, it must be the case that S^* produces two successful openings Σ_1, Σ_2 to the first-phase commit. At least one of these is different from Σ^*, yet both must satisfy $F(\Sigma_i) = Y = F(\Sigma^*)$. With probability at least $1/2$, we output $\Sigma_i \neq \Sigma^*$ as x'. Now, conditioned on all this, we argue that we had nonnegligible probability (at least $(1/2) \cdot 1/4p(n)$) of outputting Σ^* as x (prior to receiving F). This follows because $p[\Sigma^* \mid C] > 1/4p(n)$. Therefore, we break the target collision resistance property with probability at least $(1/4p(n)) \cdot (1/2) \cdot (1/2) \cdot (1/4p(n))$, which is a contradiction. □

Now we turn to the complement of E_1 in case $P = 2$, namely the event

$$E' = \{[(Y \neq F(\Sigma^*)) \vee (p[\Sigma^* \mid C] \leq 1/4p(n))] \wedge [P = 2]\},$$

Since we are now restricted to $P = 2$, there is a single first-phase decommitment value produced by S^*, which we denote by the random variable Σ.

First we argue that it is almost always the case in E' that $\Sigma \neq \Sigma^*$ (assuming S^* succeeds).

Claim 7.16. *For the event*

$$E_3 = E' \wedge (\Sigma = \Sigma^*),$$

we have $\Pr[S^* \text{ succeeds} \wedge E_3] \leq 1/4p(n)$.

Proof. [Proof of Claim] In E', we either have $Y \neq F(\Sigma^*)$, in which S^* cannot succeed unless $\Sigma \neq \Sigma^*$, or we have $p[\Sigma^* \mid C] \leq (1/4(p(n)))$, in which case S^* successfully opens to value Σ^* with probability at most $1/4p(n)$. □

So now, instead of E', we can focus on the event that $\{[\Sigma \neq \Sigma^*] \wedge [P = 2]\}$. For this, we have two cases, depending on whether the transcript T of the first-phase commitment (including the reveal) gives a binding second phase or not.

Claim 7.17. *For the event*

$$E_4 = \{[\Sigma \neq \Sigma^*] \wedge [P = 2] \wedge [\mathrm{T} \in \mathcal{B}]\},$$

we have

$$\Pr[S^* \text{ succeeds} \wedge E_4] \leq 1/4p(n).$$

Proof. [Proof of Claim] If $\mathrm{T} \in \mathcal{B}$, then the second-phase commitment is binding. Since $P = 2$, S^* can only succeed with negligible probability. □

Claim 7.18. *For the event*

$$E_5 = \{[\Sigma \neq \Sigma^*] \wedge [P = 2] \wedge [\mathrm{T} \notin \mathcal{B}]\},$$

we have

$$\Pr[S^* \text{ } \textit{succeeds} \wedge E_5] \leq 1/4p(n).$$

Proof. [Proof of Claim] Assume for contradiction that $\Pr[S^* \text{ succeeds} \wedge E_5] > 1/4p(n)$. By Markov bound, this implies that with probability at least $1/8p(n)$ over $c \overset{\mathrm{R}}{\leftarrow} C$, it holds that

$$\Pr[S^* \text{ succeeds} \wedge E_5 | C = c] > 1/8p(n). \tag{7.4}$$

We will use this to break the $\binom{2}{1}$-binding of (S, R). Similarly to the proof of Claim 7.15, we carry out two executions of (S^*, R) beginning with the same first-phase commit c. Assume that c satisfies (7.4). Then, with some probability $q[c]$ greater than $1/8p(n)$, the first execution will produce an accepting full transcript whose first phase is not in $\mathcal{B}$, with an opening to some value $\sigma \neq \sigma^* = \sigma^*[c]$. Note that conditioned on $C = c$, the probability that a random execution produces an accepting full transcript with an opening to some $\sigma' \neq \sigma$ must also be at least $q[c]$, since σ is no more likely than $\sigma^*[c]$. It follows that with probability at least $(1/8p(n) \cdot q[c] \cdot q[c] = \Omega(1/p(n)^3)$, we can output two transcripts λ and $\widetilde{\lambda}$ such that the following hold:

- both transcripts are accepting and start with c,
- the part of the first-phase commitment in both transcripts is not in $\mathcal{B}$,
- the first-phase commitment in λ is decommitted to a different value than the one in $\widetilde{\lambda}$.

Namely, we break the $\binom{2}{1}$-binding of (S, R) with nonnegligible probability. □

We the above claims, we complete the proof. By inspection, we have $\Pr[\bigvee_i E_i] = 1$, and thus:

$$\Pr[S^* \text{ succeeds}] \leq \Pr[E_1] + \sum_{i=1}^{4} \Pr[S^* \text{ succeeds} \wedge E_i] \leq \frac{1}{2} + \frac{1}{p(n)},$$

as desired. □

Boosting the binding. The commitment scheme $(\mathbb{S}, \mathbb{R})$ from Lemma 7.13 is only $(\frac{3}{4} + \mathrm{neg}(n))$-binding. Nonetheless, by the following "folklore" claim, $(\mathbb{S}, \mathbb{R})$ implies a commitment scheme that is $\mathrm{neg}(n)$-binding and preserves the same hiding property as the original scheme.

Claim 7.19. *There exists an efficient procedure that for any function* $\delta \geq 1/\mathrm{poly}(n)$ *converts a statistically [resp., computationally]* $(1 - \delta(n))$*-binding commitment scheme* $(\mathbb{S}, \mathbb{R})$ *into a commitment scheme* (S, R) *that is statistically [resp., computationally] binding. Furthermore, if* $(\mathbb{S}, \mathbb{R})$ *is statistically [resp., computationally] hiding, so is* (S, R).

Proof. The protocol (S, R) is defined as follows: in order to commit to a bit b, the two parties run $t = \lceil n/\delta \rceil = \mathrm{poly}(n)$ independent executions of the commit stage of $(\mathbb{S}(b), \mathbb{R})$ one after the other, where S and R acting as $\mathbb{S}$ and $\mathbb{R}$ respectively. In the reveal stage, S decommits, via the reveal stage of $(\mathbb{S}, \mathbb{R})$, all the t commitments and R accepts if and only if all the commitments are opened successfully to the same value. The hiding of the above scheme follows by a straightforward hybrid argument. For the binding part, let S^* be a PPT trying to break the binding of (S, R). We show that S^* breaks the binding of (S, R) only with negligible probability, and since S^* was arbitrarily chosen it follows that (S, R) is computationally binding.

We say that S^* ***breaks the binding of the i^{th} execution*** of $(\mathbb{S}, \mathbb{R})$ if while trying to break the binding of (S, R) it successfully opens the i^{th} commitment into two different values. Notice that this event depends on several random variables: $C_{<i}$, the coins of S^* and the coins of R in the first $i-1$ executions; C_i, the coins of R in the i'th execution; and $C_{>i}$, the coins of R in executions $i+1, \ldots, t$. For settings $(c_{<i}, c_i) \in \mathrm{Supp}(C_{<i}, C_i)$, we define $q_i(c_{<i}, c_i)$ to be the probability over $C_{>i}$ that S^* breaks the binding of the i^{th} execution conditioned on $(C_{<i}, C_i) = (c_{<i}, c_i)$.

For an arbitrary positive polynomial p, define a prefix $c_{<i}$ to ***bad*** if $\Pr[q_i(c_{<i}, C_i) > 1/p(n)] > 1 - \delta + 1/p(n)$, and otherwise call $c_{<i}$ ***good***. We will now show that $\Pr[C_{<i} \text{ is bad}] \leq 1/p(n)$. Suppose not. Then we can construct an efficient algorithm $\mathbb{S}^*$ that breaks the binding of $(\mathbb{S}, \mathbb{R})$ with probability $1 - \delta + 1/3p(n)$. In the commit stage, $\mathbb{S}^*$ first finds a value $c_{<i}$ for which $\Pr[q_i(c_{<i}, C_i) > 1/2p(n)] > 1 - \delta + 1/2p(n)$ and "hardwires" this value into S^*. Note that the above can be done efficiently and with overwhelming success probability by random sampling, given oracle access to S^*.[13]

When interacting with $\mathbb{R}$, $\mathbb{S}^*$ acts as S^* does in the ith execution of (S^*, R). With probability at least $1 - \delta + 1/2p(n)$ over the coins c_i of $\mathbb{R}$, we have $q_i(c_{<i}, c_i) > 1/2p(n)$. If this occurs, then by randomly continuing the simulation of (S^*, R) with $O(n \cdot p(n))$ independent choices of $C_{>i}$, $\mathbb{S}^*$ will be able to break the binding with probability $1 - \mathrm{neg}(n)$. Thus, $\mathbb{S}^*$ breaks the binding of $(\mathbb{S}, \mathbb{R})$ with probability $1 - \delta + 1/2p(n) - \mathrm{neg}(n) > 1 - \delta + 1/3p(n)$.

Let E_1 be the event that for some i, $C_{<i}$ is bad. By the above and a union bound, $\Pr[E_1] \leq t/p(n)$. Let E_2 be the event that for some i, $q_i(C_{<i}, C_i) \leq 1/p(n)$ but S^* breaks the binding of the i'th execution. By the definition of q_i, we have $\Pr[E_2] \leq t/p(n)$. Finally, we have

$$
\begin{aligned}
&\Pr[S^* \text{ breaks the binding} \wedge \neg E_1 \wedge \neg E_2] \\
&\leq \Pr\left[\bigwedge_{i=1}^{t} [(C_{<i} \text{ good}) \wedge (q_i(C_{<i}, C_i) > 1/p(n))]\right] \\
&= \prod_{i=1}^{t} \Pr\left[(C_{<i} \text{ good}) \wedge (q_i(C_{<i}, C_i) > 1/p(n)) \,\middle|\, \bigwedge_{j<i} [(C_{<j} \text{ good}) \wedge (q_j(C_{<j}, C_j) > 1/p(n))]\right] \\
&\leq (1 - \delta + 1/p(n))^t \\
&= \mathrm{neg}(n) + t/p(n),
\end{aligned}
$$

where the last inequality can be seen by considering any fixed value $C_{<i} = c_{<i}$, which fixes the event on which we are conditioning in the i'th factor and whether $C_{<i}$ is

[13] For any given value of $c_{<i}$, we can evaluate, with save but negligible error probability, the value of $\Pr[q_i(c_{<i}, C_i) > 1/2p(n)]$ up to an additive error $1/4p(n)$. (In order to do so, we sample $n \cdot p(n)$ uniformly chosen continuations of the commit and then reveal stage of (S^*, R), and computes the fraction of continuations in which S^* breaks the binding of the i'th commitment). Thus, we can find a value of $c_{<i}$ for which $\Pr[q_i(c_{<i}, C_i) > 1/2p(n)] > 1 - \delta + 1/2p(n)$, by sampling $n \cdot p(n)$ values for $c_{<i}$, and for each of these values evaluate $\Pr[q_i(c_{<i}, C_i) > 1/2p(n)]$.

good or bad. If $c_{<i}$ is bad, then the probability in the i'th factor is 0. If $c_{<i}$ is good, then the probability (over just C_i) is at most $(1 - \delta + 1/p(n))$ by the definition of good. Taking $p(n)$ to be an arbitrarily large polynomial, we deduce that S^* breaks the binding with negligible probability. □

Having established the appropriate claims and lemmas, we now state what is achievable from our transformation.

Theorem 7.20. *There exists an efficient procedure, call it 2-to-1-*$\mathsf{FullTransform}$*, that takes as input a security parameter* 1^n*, a two-phase commitment scheme* (S, R) *with message lengths* $(k_1, k_2) = (n, 1)$*, and a family of functions* $\mathcal{F} = \bigcup_n \mathcal{F}_n = \{f \colon \{0,1\}^n \to \{0,1\}^m\}$*, and outputs a commitment scheme*
$(S, R) =$ *2-to-1-*$\mathsf{FullTransform}((\mathsf{S}, \mathsf{R}), \mathcal{F})$ *satisfying the following properties:*

- *If* (S, R) *is statistically hiding and* $\mathcal{F}$ *has the large preimages property, then* (S, R) *is statistically hiding.*
- *If* (S, R) *is statistically [resp., computationally]* $\binom{2}{1}$*-binding and* $\mathcal{F}$ *has statistical [resp., computational] target collision resistance, then* (S, R) *is statistically [resp., computationally] binding (in the standard sense of binding).*
- *If* (S, R) *is public coin, then* (S, R) *is also public coin.*

Proof. We describe the 2-to-1-$\mathsf{FullTransform}$ algorithm, recapping what we have done thus far, as follows.

1. Apply Algorithm 7.1 on (S, R) and $\mathcal{F}$ to obtain a (standard) commitment scheme $(\mathbb{S}, \mathbb{R})$. Lemmas 7.10 and 7.13 state that for the right properties of both $(\mathsf{S}', \mathsf{R}')$ and $\mathcal{F}$ (see the first two items in 7.20 above), $(\mathbb{S}, \mathbb{R})$ is hiding and $(1/2 + \text{neg}(n))$-binding.
2. Next, using Claim 7.19, boost the binding of $(\mathbb{S}, \mathbb{R})$ to obtain a scheme (S, R) that is $\text{neg}(n)$-binding while not affecting the hiding property. Output (S, R) as our desired scheme.

As for the preservation of the public coin property, observe that the messages sent by $\mathbb{R}$ that are specific to the 2-to-1-$\mathsf{Transform}$ are choosing $f \leftarrow \mathcal{F}$ and selecting *phase* $\leftarrow \{0, 1\}$, both of which are public coin operations. □

8. Putting it Together. Now, we put together everything from the previous sections to establish our main theorem.

Restatement of Theorem 2 (1.1). *Given a one-way function* $f : \{0,1\}^n \to \{0,1\}^n$*, we can construct in time polynomial in* n *a public-coin commitment scheme* $(\mathbb{S}, \mathbb{R})$ *that is statistically hiding and computationally binding.*

The statistical hiding property holds regardless of whether or not f *is secure (hard to invert). On the other hand, if* f *is uniformly (resp., nonuniformly) secure, than* $(\mathbb{S}, \mathbb{R})$ *will be computationally binding with uniform (resp., nonuniform) security.*

Proof. [Proof of Theorem 1.1] We start off by constructing a collection of two-phase commitment schemes from f using Theorem 6.1. For any polynomial $k(n)$ (which we will choose below), we can construct in time polynomial in n a collection of $m = \text{poly}(n)$ public-coin two-phase commitment schemes $\mathcal{COM} = \{\mathsf{Com}_1, \cdots, \mathsf{Com}_m\}$ with message lengths $(k(n), 1)$ such that:

- there exists an index $i \in \{1, 2, \ldots, m\}$ such that scheme Com_i is statistically hiding, and
- for every index $i \in \{1, 2, \ldots, m\}$, scheme Com_i is computationally $\binom{2}{1}$ binding.

(As remarked after Theorem 6.1, we can obtain two-phase commitments with message lengths $(k(n), k(n))$ for any polynomial k that we choose. Using only 1 bit of the 2nd-phase message (padding with $k - 1$ zeroes), we obtain message lengths $(k, 1)$.)

Now in order to apply Theorem 7.20, from f we use [7, 29] to obtain a universal one-way hash family $\mathcal{F}_n = \{f \colon \{0,1\}^{k(n)} \to \{0,1\}^{k(n)/2}\}$ for some polynomial k (which we use to determine the message length for the 2-phase commitment above).[14] Let the resulting (standard) commitment schemes be $\mathsf{Com}'_i = \text{2-to-1-}\mathsf{FullTransform}(\mathsf{Com}_i, \mathcal{F})$. By Theorem 7.20 and Lemma 7.9, we know that:

- Com'_i is statistically hiding if Com_i is statistically hiding,
- Com'_i is computationally binding if Com_i is computationally $\binom{2}{1}$-binding, and
- Com'_i is public coin if Com_i is public coin.

This means that we now have a collection of public-coin (standard) commitment schemes $\mathcal{COM}' = \{\mathsf{Com}'_1, \cdots, \mathsf{Com}'_m\}$, where $m = \text{poly}(n)$, such that:

- there exists an index $i \in \{1, 2, \ldots, m\}$ such that scheme Com'_i is statistically hiding, and
- for every index $i \in \{1, 2, \ldots, m\}$, scheme Com'_i is computationally binding (in the standard sense of binding).

We are almost done, except that we are still left with a collection of commitments instead of a *single* commitment scheme. The following claim states that the latter collection can be converted into the desired commitment scheme.

> Claim 8.1. *There is an efficient procedure that converts a polynomial collection of commitment schemes, at least one of which is statistically hiding and all are computationally binding, into a* single *commitment scheme that is statistically hiding and computationally binding. In addition, if we start off with public-coin schemes, we also end up with a public-coin scheme.*
>
> *Proof.* To commit to a bit b, we randomly secret-share $b = b_1 \oplus \cdots \oplus b_m$ and commit to share b_i using the i'th commitment scheme. Alternatively, the proposition can be deduced from [25, Thm. 5.2]. □

The main theorem statement is now complete since we now have a *single* commitment scheme that is statistically hiding and computationally binding, and the only complexity assumption made is the existence of one-way functions.

We now proceed to the additional properties mentioned. By inspection, we observe that the statistical hiding properties throughout the construction hold regardless of the security of f (see e.g. Lemma 6.13). As for nonuniform security, we observe that our construction is "fully black-box" in the sense of [40]; in particular, the computational binding property is proven by specifying for every polynomial p, a PPT reduction R such that if $\mathbb{S}^*$ is *any* sender strategy (of arbitrary complexity) that breaks the binding property with probability $1/p(n)$, then $R^{\mathbb{S}^*}$ inverts f with nonnegligible probability. In particular, if $\mathbb{S}^*$ a nonuniform PPT algorithm, then we obtain a nonuniform PPT inverter for f, which cannot exist if f is nonuniformly secure. □

Acknowledgments. We are grateful to the anonymous referees for their extensive comments and corrections. We also thank Oded Goldreich for helpful suggestions.

[14] Since we are using here the uniform definition of universal one-way hash family (i.e., where x is sampled by A), we need to use the theorem of Katz and Koo [29]. In their theorem, however, it is not explicitly defined whether or not the adversary can encode additional information (i.e. state) between the declaration of x and finding the collision (see Remark 7.5). Fortunately, the stronger version of this theorem required by our proof, follows readily from their original proof.

REFERENCES

[1] Nguyen, M., Ong, S. J., and Vadhan, S., “Statistical Zero-Knowledge Arguments for NP from Any One-Way Function,” in *Proceedings of the 47th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 3–14, IEEE Computer Society, 2006.

[2] Haitner, I. and Reingold, O., “Statistically-Hiding Commitment from Any One-Way Function,” in *Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC)*, ACM Press, 2007.

[3] Shannon, C., “Communication Theory of Secrecy Systems,” *Bell System Technical Journal*, vol. 28, no. 4, 1949, pp. 656–715.

[4] Diffie, W. and Hellman, M. E., “New Directions in Cryptography,” *IEEE Transactions on Information Theory*, vol. 22, no. 6, 1976, pp. 644–654.

[5] Håstad, J., Impagliazzo, R., Levin, L. A., and Luby, M., “A pseudorandom generator from any one-way function,” *SIAM Journal on Computing*, vol. 28, no. 4, 1999, pp. 1364–1396. Preliminary versions in *STOC’89* and *STOC’90*.

[6] Goldreich, O., Goldwasser, S., and Micali, S., “How to Construct Random Functions,” *Journal of the ACM*, vol. 33, no. 4, 1986, pp. 792–807.

[7] Rompel, J., “One-Way Functions are Necessary and Sufficient for Secure Signatures,” in *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing (STOC)*, pp. 387–394, 1990.

[8] Naor, M., “Bit Commitment Using Pseudorandomness,” *Journal of Cryptology*, vol. 4, no. 2, 1991, pp. 151–158. Preliminary version in *CRYPTO’89*.

[9] Goldreich, O., Micali, S., and Wigderson, A., “Proofs that Yield Nothing But Their Validity or All Languages in NP Have Zero-Knowledge Proof Systems,” *Journal of the ACM*, vol. 38, no. 1, 1991, pp. 691–729. Preliminary version in *FOCS’86*.

[10] Impagliazzo, R. and Luby, M., “One-way Functions are Essential for Complexity Based Cryptography,” in *Proceedings of the 30th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 230–235, 1989.

[11] Ostrovsky, R. and Wigderson, A., “One-Way Functions are Essential for Non-Trivial Zero-Knowledge,” in *Proceedings of the 2nd Israel Symposium on Theory of Computing Systems*, pp. 3–17, IEEE Computer Society, 1993.

[12] Impagliazzo, R. and Rudich, S., “Limits on the provable consequences of one-way permutations,” in *Proceedings of the 21st Annual ACM Symposium on Theory of Computing (STOC)*, pp. 44–61, ACM Press, 1989.

[13] Simon, D., “Finding Collisions on a One-Way Street: Can Secure Hash Functions Be Based on General Assumptions?,” in *Advances in Cryptology – EUROCRYPT ’98*, vol. 1403 of *Lecture Notes in Computer Science*, pp. 334–345, Springer, 1998.

[14] Brassard, G., Chaum, D., and Crépeau, C., “Minimum Disclosure Proofs of Knowledge,” *Journal of Computer and System Sciences*, vol. 37, no. 2, 1988, pp. 156–189.

[15] Brassard, G., Crépeau, C., and Yung, M., “Constant-round perfect zero-knowledge computationally convincing protocols,” *Theoretical Computer Science*, vol. 84, no. 1, 1991, pp. 23–52.

[16] Naor, M., Ostrovsky, R., Venkatesan, R., and Yung, M., “Perfect Zero-Knowledge Arguments for NP Using Any One-Way Permutation,” *Journal of Cryptology*, vol. 11, no. 2, 1998, pp. 87–108. Preliminary version in *CRYPTO’92*.

[17] Goldreich, O., Micali, S., and Wigderson, A., “How to Play any Mental Game or A Completeness Theorem for Protocols with Honest Majority,” in *Proceedings of the 19th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 218–229, ACM Press, 1987.

[18] Boyar, J. F., Kurtz, S. A., and Krentel, M. W., “A Discrete Logarithm Implementation of Perfect Zero-Knowledge Blobs,” *Journal of Cryptology*, vol. 2, no. 2, 1990, pp. 63–76.

[19] Chaum, D., Damgård, I., and Graaf, J., “Multiparty Computations Ensuring Privacy of Each Party’s Input and Correctness of the Result.,” in *Advances in Cryptology – CRYPTO ’87*, vol. 293 of *Lecture Notes in Computer Science*, pp. 87–119, Springer, 1987.

[20] Pedersen, T. P., “Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing,” in *Advances in Cryptology – CRYPTO ’91*, vol. 576 of *Lecture Notes in Computer Science*, pp. 129–140, Springer, 1991.

[21] Goldwasser, S., Micali, S., and Rivest, R. L., “A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks,” *SIAM Journal on Computing*, vol. 17, no. 2, 1988, pp. 281–308. Preliminary version in *FOCS’84*.

[22] Goldreich, O. and Kahan, A., “How to Construct Constant-Round Zero-Knowledge Proof Systems for NP,” *Journal of Cryptology*, vol. 9, no. 3, 1996, pp. 167–190.

[23] Naor, M. and Yung, M., “Universal One-Way Hash Functions and their Cryptographic Appli-

cations," in *Proceedings of the 21st Annual ACM Symposium on Theory of Computing (STOC)*, pp. 33–43, ACM Press, 1989.

[24] Damgård, I. B., Pedersen, T. P., and Pfitzmann, B., "Statistical secrecy and multibit commitments," *IEEE Transactions on Information Theory*, vol. 44, no. 3, 1998, pp. 1143–1151.

[25] Haitner, I., Horvitz, O., Katz, J., Koo, C., Morselli, R., and Shaltiel, R., "Reducing Complexity Assumptions for Statistically-Hiding Commitment," in *Advances in Cryptology – EUROCRYPT 2005*, vol. 3494 of *Lecture Notes in Computer Science*, pp. 58–77, Springer, 2005.

[26] Ong, S. J. and Vadhan, S., "Zero Knowledge and Soundness are Symmetric," in *Advances in Cryptology – EUROCRYPT 2007*, vol. 4515 of *Lecture Notes in Computer Science*, pp. 187–209, Springer, 2007.

[27] Ostrovsky, R., "One-Way Functions, Hard on Average Problems, and Statistical Zero-Knowledge Proofs," in *Proceedings of the 6th Annual Structure in Complexity Theory Conference*, pp. 133–138, IEEE Computer Society, 1991.

[28] Nguyen, M. and Vadhan, S., "Zero Knowledge with Efficient Provers," in *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 287–295, ACM Press, 2006.

[29] Katz, J. and Koo, C., "On Constructing Universal One-Way Hash Functions from Arbitrary One-Way Functions," Cryptology ePrint Archive, 2005.

[30] Haitner, I., Reingold, O., Vadhan, S., and Wee, H., "Inaccessible Entropy," in *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC)*, ACM Press, 2009. To appear.

[31] Goldreich, O., *Foundations of Cryptography: Basic Tools.* Cambridge University Press, 2001.

[32] Ostrovsky, R., Venkatesan, R., and Yung, M., "Fair Games Against an All-Powerful Adversary," *AMS DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 1993, pp. 155–169. Preliminary version in *SEQUENCES'91*.

[33] Haitner, I. and Reingold, O., "A New Interactive Hashing Theorem," in *Proceedings of the 22th Annual IEEE Conference on Computational Complexity*, IEEE Computer Society, 2007.

[34] Cachin, C., Crépeau, C., and Marcil, J., "Oblivious Transfer with a Memory-Bounded Receiver," in *Proceedings of the 39th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 493–502, IEEE Computer Society, 1998.

[35] Ding, Y. Z., Harnik, D., Rosen, A., and Shaltiel, R., "Constant-Round Oblivious Transfer in the Bounded Storage Model," in *Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004*, vol. 2951 of *Lecture Notes in Computer Science*, pp. 446–472, Springer, 2004.

[36] Crépeau, C. and Savvides, G., "Optimal Reductions Between Oblivious Transfers Using Interactive Hashing," in *Advances in Cryptology – EUROCRYPT 2006*, vol. 4004 of *Lecture Notes in Computer Science*, pp. 201–221, Springer, 2006.

[37] Bennett, C. H., Brassard, G., and Robert, J.-M., "Privacy amplification by public discussion," *SIAM Journal on Computing*, vol. 17, no. 2, 1988, pp. 210–229.

[38] Goldreich, O., Goldwasser, S., and Linial, N., "Fault-Tolerant Computation in the Full Information Model.," *SIAM Journal on Computing*, vol. 27, no. 2, 1998, pp. 506–544.

[39] Ong, S. J. and Vadhan, S., "An Equivalence between Zero Knowledge and Commitments," in *Proceedings of the Third Theory of Cryptography Conference (TCC '08)* (Canetti, R., ed.), vol. 4948 of *Lecture Notes in Computer Science*, pp. 482–500, Springer-Verlag, 19–21 March 2008.

[40] Reingold, O., Trevisan, L., and Vadhan, S., "Notions of Reducibility between Cryptographic Primitives," in *Proceedings of the First Theory of Cryptography Conference (TCC '04)* (Naor, M., ed.), vol. 2951 of *Lecture Notes in Computer Science*, pp. 1–20, Springer-Verlag, 19–21 February 2004.

Appendix A. Collision Probability Lemmas.

We prove the lemmas presented in Section 6.2.1.

Restatement of Lemma 1 (6.4). *For independent pairs of random variables* $(X_1, Y_1), \ldots, (X_m, Y_m)$,

$$\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) = \prod_{i=1}^{m} \mathrm{CP}^{1/2}(X_i|Y_i) \ .$$

Note that X_i *and* Y_i *can be correlated, it is only required that the pair* (X_i, Y_i) *be independent from the other tuples.*

Proof. Since the X_i's are independent, for all $y_1, \ldots, y_m$, we have

$$\mathrm{CP}((X_1, \ldots, X_m)|_{Y_1=y_1,\ldots,Y_m=y_m}) = \prod_{i=1}^{m} \mathrm{CP}(X_i|_{Y_i=y_i}) \ . \tag{A.1}$$

This gives us

$$\begin{aligned}
&\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) \\
&= \mathop{\mathrm{E}}_{(Y_1,\ldots,Y_m)} \left[\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|_{Y_1,\ldots,Y_m})\right] \\
&= \mathop{\mathrm{E}}_{(Y_1,\ldots,Y_m)} \left[\prod_{i=1}^{m} \mathrm{CP}^{1/2}(X_i|_{Y_i})\right] && \text{(by A.1)} \\
&= \prod_{i=1}^{m} \mathop{\mathrm{E}}_{Y_i} \left[\mathrm{CP}^{1/2}(X_i|_{Y_i})\right] && \text{(by independence of } Y_i\text{'s)} \\
&= \prod_{i=1}^{m} \mathrm{CP}^{1/2}(X_i|Y_i) \ .
\end{aligned}$$

□

Restatement of Lemma 2 (6.5). *Suppose random variables* $(X_1, Y_1), \ldots, (X_m, Y_m)$ *satisfy the following conditions for some values of* $\alpha_1, \ldots, \alpha_m \in \mathbb{R}^+$ *and all* $i = 1, 2, \ldots, m$*:*

1. *For every* $(y_1, \ldots, y_{i-1}) \in \mathrm{Supp}(Y_1, Y_2, \ldots, Y_{i-1})$,

$$\mathrm{CP}^{1/2}(X_i|_{Y_1=y_1,\ldots,Y_{i-1}=y_{i-1}} \mid Y_i|_{Y_1=y_1,\ldots,Y_{i-1}=y_{i-1}}) \leq \alpha_i \ .$$

2. *For every* $(y_1, \ldots, y_i) \in \mathrm{Supp}(Y_1, Y_2, \ldots, Y_i)$*, the* $i+1$ *random variables* $X_1, X_2, \ldots, X_i$*, and* Y_{i+1} *are independent after conditioning on* $Y_1 = y_1, \ldots, Y_i = y_i$*.*

Then,

$$\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) \leq \prod_{i=1}^{m} \alpha_i \ .$$

Proof. By induction, it suffices to prove

$$\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) \leq \alpha_m \cdot \mathrm{CP}^{1/2}\left((X_1, \ldots, X_{m-1})|(Y_1, \ldots, Y_{m-1})\right) \ , \tag{A.2}$$

and then by iteratively expanding $\mathrm{CP}^{1/2}\left((X_1, \ldots, X_{m-1})|(Y_1, \ldots, Y_{m-1})\right)$ in terms of α_j's, we get our result. To simplify notation, we write $X'_m = X_m|_{Y_1=y_1,\ldots,Y_{m-1}=y_{m-1}}$ and $Y'_m = Y_m|_{Y_1=y_1,\ldots,Y_{m-1}=y_{m-1}}$ when $y_1, \ldots, y_{m-1}$ are clear from context. We prove

(A.2) as follows:

$$
\begin{aligned}
&\mathrm{CP}^{1/2}((X_1,\ldots,X_m)|(Y_1,\ldots,Y_m)) && \text{(A.3)}\\
&= \mathop{\mathrm{E}}_{(Y_1,\ldots,Y_m)}\left[\mathrm{CP}^{1/2}((X_1,\ldots,X_m)|_{Y_1,\ldots,Y_m})\right] && \text{(A.4)}\\
&= \mathop{\mathrm{E}}_{(Y_1,\ldots,Y_{m-1})}\left[\mathop{\mathrm{E}}_{Y'_m}\left[\mathrm{CP}^{1/2}((X_1,\ldots,X_m)|_{Y_1,\ldots,Y'_m})\right]\right] && \text{(A.5)}\\
&= \mathop{\mathrm{E}}_{(Y_1,\ldots,Y_{m-1})}\left[\mathop{\mathrm{E}}_{Y'_m}\left[\mathrm{CP}^{1/2}((X_1,\ldots,X_{m-1})|_{Y_1,\ldots,Y'_m})\cdot \mathrm{CP}^{1/2}(X_m|_{Y_1,\ldots,Y'_m})\right]\right] && \text{(A.6)}\\
&= \mathop{\mathrm{E}}_{(Y_1,\ldots,Y_{m-1})}\left[\mathrm{CP}^{1/2}((X_1,\ldots,X_{m-1})|_{Y_1,\ldots,Y_{m-1}})\cdot \mathop{\mathrm{E}}_{Y'_m}\left[\mathrm{CP}^{1/2}(X_m|_{Y_1,\ldots,Y'_m})\right]\right] && \text{(A.7)}\\
&= \mathop{\mathrm{E}}_{(Y_1,\ldots,Y_{m-1})}\left[\mathrm{CP}^{1/2}((X_1,\ldots,X_{m-1})|_{Y_1,\ldots,Y_{m-1}})\cdot \mathrm{CP}^{1/2}(X'_m|Y'_m)\right] && \text{(A.8)}\\
&\le \alpha_m\cdot \mathop{\mathrm{E}}_{(Y_1,\ldots,Y_{m-1})}\left[\mathrm{CP}^{1/2}((X_1,\ldots,X_{m-1})|_{Y_1,\ldots,Y_{m-1}})\right] && \text{(A.9)}\\
&\le \alpha_m\cdot \mathrm{CP}^{1/2}\left((X_1,\ldots,X_{m-1})|(Y_1,\ldots,Y_{m-1})\right)\ . && \text{(A.10)}
\end{aligned}
$$

Equation (A.6) follows because $X_1,\ldots,X_m$ conditioned on $Y_1=y_1,\ldots,Y_m=y_m$ are independent. Equation (A.7) follows because $X_1,\ldots,X_{m-1}$, and Y_m conditioned on $Y_1=y_1,\ldots,Y_{m-1}=y_{m-1}$ are independent. Finally, (A.9) follows from the assumption that for all $(y_1,\ldots,y_{i-1})\in\mathrm{Supp}(Y_1,Y_2,\ldots,Y_{m-1})$,

$$\mathrm{CP}^{1/2}(X'_m|Y'_m)=\mathrm{CP}^{1/2}(X_m|_{Y_1=y_1,\ldots,Y_{m-1}=y_{m-1}}\mid Y_m|_{Y_1=y_1,\ldots,Y_{m-1}=y_{m-1}})\le\alpha_m\ .$$

□

Restatement of Lemma 3 (6.6). *(Randomness Extraction Lemma.) Let (X,Y) be any (possibly correlated) pair of random variables, and let random variable H denote a random hash function from a family of pairwise-independent hash functions $\mathcal{H}$ with range $\{0,1\}^\alpha$. Suppose the hash functions from $\mathcal{H}$ are represented by $(q-\alpha)$-bit strings and* $\mathrm{CP}^{1/2}(X|Y)\le\sqrt{2^{-(\alpha+3)}}$. *If H is independent from (X,Y), then*

$$\mathrm{CP}^{1/2}((H,H(X))|Y)\le\sqrt{2^{-(q-1)}}\ .$$

Proof. We bound the value of $\mathrm{CP}^{1/2}((H,H(X))|Y)$ as follows:

$$\begin{aligned}
&\mathrm{CP}^{1/2}(H,H(X)|Y)\\
&= \mathop{\mathrm{E}}_{y\leftarrow Y}\left[\mathrm{CP}^{1/2}(H,H(X)|_{Y=y})\right]\\
&\leq \mathop{\mathrm{E}}_{y\leftarrow Y}\left[\mathrm{CP}^{1/2}(H)\cdot\sqrt{\mathrm{CP}(X|_{Y=y})+2^{-\alpha}}\right] && \left(\begin{array}{c}\text{since } \mathrm{CP}(H,H(Z))\leq\\ \mathrm{CP}(H)\cdot(\mathrm{CP}(Z)+2^{-\alpha})\end{array}\right)\\
&\leq \mathop{\mathrm{E}}_{y\leftarrow Y}\left[\mathrm{CP}^{1/2}(H)\cdot\left(\mathrm{CP}^{1/2}(X|_{Y=y})+\sqrt{2^{-\alpha}}\right)\right] && \text{(Cauchy-Schwarz/Jensen)}\\
&= \mathrm{CP}^{1/2}(H)\cdot\left(\left(\mathop{\mathrm{E}}_{y\leftarrow Y}\left[\mathrm{CP}^{1/2}(X|_{Y=y})\right]\right)+\sqrt{2^{-\alpha}}\right)\\
&= \mathrm{CP}^{1/2}(H)\cdot(\mathrm{CP}^{1/2}(X|Y)+\sqrt{2^{-\alpha}})\\
&= \sqrt{2^{-(q-\alpha)}}\cdot(\mathrm{CP}^{1/2}(X|Y)+\sqrt{2^{-\alpha}}) && \text{(since } |h|=q-\alpha)\\
&\leq \sqrt{2^{-(q-\alpha)}}\cdot\left(\sqrt{\frac{2^{-\alpha}}{8}}+\sqrt{2^{-\alpha}}\right)\\
&< \sqrt{2^{-(q-\alpha)}}\cdot\left(\sqrt{2^{-\alpha}}\cdot\sqrt{2}\right)\\
&= \sqrt{2^{-(q-1)}}\ .
\end{aligned}$$

□

Restatement of Lemma 4 (6.7). *For any triple of (possibly correlated) random variables X, Y and Z,*

$$\mathrm{CP}^{1/2}(X|Y)\leq\mathrm{CP}^{1/2}(X|(Y,Z))\leq\sqrt{|\mathrm{Supp}(Z)|}\cdot\mathrm{CP}^{1/2}(X|Y)\ .$$

Proof. For each $y\in\mathrm{Supp}(Y)$ and $z\in\mathrm{Supp}(Z)$, let $v_{y,z}\in\mathbb{R}^{\mathrm{Supp}(X)}$ be the vector whose x'th entry is $v_{y,z}(x)=\Pr[X=x\wedge Z=z|Y=y]$. With this, we compute for each y:

$$\begin{aligned}
\left\|\sum_z v_{y,z}\right\|_2 &\leq \sum_z\|v_{y,z}\|_2 && \text{(triangle inequality)}\\
&\leq\sqrt{\mathrm{Supp}(Z)}\cdot\left(\sum_z\|v_{y,z}\|_2^2\right)^{1/2} && \text{(Cauchy-Schwarz/Jensen)}\\
&=\sqrt{\mathrm{Supp}(Z)}\cdot\left(\sum_{z,x}v_{y,z}(x)^2\right)^{1/2}\\
&\leq\sqrt{\mathrm{Supp}(Z)}\cdot\left(\sum_x\left(\sum_z v_{y,z}(x)\right)^2\right)^{1/2} && \text{(nonnegativity of } v_{y,z}(x))\\
&=\sqrt{\mathrm{Supp}(Z)}\cdot\left\|\sum_z v_{y,z}\right\|_2\ .
\end{aligned}$$

Since $\mathrm{CP}^{1/2}(X|_{Y=y})=\|\sum_z v_{y,z}\|_2$ and $\mathrm{CP}^{1/2}\left((X|_{Y=y})\,|\,(Z|_{Y=y})\right)=\sum_z\|v_{y,z}\|_2$, taking expectations over Y for both sides of the above inequalities yields the claim. □

Restatement of Lemma 5 (6.8). *Let random variable H denote a random hash function from a family of pairwise-independent hash functions $\mathcal{H}$ with range $\{0,1\}^{\alpha}$. For any $\varepsilon > 0$, if $\mathrm{CP}(X) \leq \varepsilon^2 \cdot 2^{-\alpha}$ and H is independent from X, then random variable $(H, H(X))$ is ε-close in statistical distance to uniform.*

Proof. Let $D = 2^{q-\alpha}$ and $L = 2^{\alpha}$. We bound the statistical distance of $(H, H(X))$ from uniform as follows:

$$\begin{aligned}
\frac{1}{2}\left|(H, H(X)) - U_q\right|_1 &\leq \frac{\sqrt{DL}}{2}\left\|(H, H(X)) - U_q\right\|_2 \\
&= \frac{\sqrt{DL}}{2} \cdot \sqrt{\mathrm{CP}(H, H(X)) - 2^{-q}} \\
&\leq \frac{\sqrt{DL}}{2} \cdot \sqrt{\frac{1}{D}\left(\mathrm{CP}(X) + \frac{1}{L}\right) - \frac{1}{DL}} \\
&= \frac{\sqrt{\mathrm{CP}(X) \cdot L}}{2} \\
&\leq \frac{\varepsilon}{2}\ .
\end{aligned}$$

□